

# Frequency Domain System Identification

## With a Rational, Multi-Band, Curve-Fitting Approach

Deborah A. Jue\*

Department of Mechanical Engineering  
University of California, Berkeley

A report submitted in partial satisfaction of the

requirements for the degree of

Master of Science

in

Mechanical Engineering

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Research Advisor:

Professor Andrew K. Packard

December 17, 1998

---

\*Supported in part by NASA Research Project NAG5-6757

Thanks to my advisor Andrew Packard, Kameshwar Poolla, Darin Fisher, Khalid Al-Ali and the rest of the Berkeley Center for Control and Identification lab for their assistance and support

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Frequency Domain System Identification</b>	<b>1</b>
2.1	Fitting Frequency Data with a Rational Function . . . . .	1
2.2	Santhanum-Koerner Algorithm . . . . .	2
2.2.1	Bilinear Transform . . . . .	2
2.2.2	The Least-Squares Equation . . . . .	3
2.2.3	Further Comments on the SK Algorithm . . . . .	4
2.3	Multi-Band Fit . . . . .	4
2.3.1	Dividing the Frequency Range into Bands . . . . .	5
2.3.2	Recommended Fitting Technique . . . . .	5
2.4	Multi-Variable Systems . . . . .	6
2.4.1	Vector case (SIMO) . . . . .	6
2.4.2	Matrix Case (MIMO) . . . . .	6
2.4.3	Realization of a MIMO System . . . . .	7
<b>3</b>	<b>Obtaining the Frequency Response Data</b>	<b>9</b>
3.1	Steady State Response and the Transfer Function . . . . .	9
3.2	Experimental Determination of the Transfer Function . . . . .	10
3.2.1	Waiting For Steady State . . . . .	10
3.2.2	A Statistical Approach, Finite Time Experiment . . . . .	11
<b>4</b>	<b>Examples</b>	<b>12</b>
4.1	Example 1: Basic Multi-Band Fitting Technique . . . . .	13
4.2	Example 2: Fitting Exact Frequency Response Data . . . . .	15
4.3	Example 3: Fitting Simulated-Experimental Frequency Response Data . . . . .	18
4.4	Example 4: Fitting Real Frequency Response Data from a Stable System . . . . .	19
<b>5</b>	<b>Conclusions</b>	<b>20</b>

## List of Figures

1	System Composed of Three Separate, Stable Systems. . . . .	12
2	Estimation Display. . . . .	13
3	Effect of Pre-Warp Term on Bilinear Transform. . . . .	14
4	Exact Frequency Response for Virtual System. . . . .	15
5	Highest Possible Stable Fit, One Band, 116 states . . . . .	16
6	Balanced Multi-Band Solution, Three Bands 236 states . . . . .	17
7	Percentage Error, One Band Fit and Multi-Band Fit. . . . .	18

8	Display During Estimation Process. . . . .	19
9	Hankel Singular Values. . . . .	20
10	Comparison of Truncated (180 states) and Full Estimate (236 states), (2,3). . . . .	21
11	Comparison of Poles, Original and Truncated System. . . . .	22
12	Actual and Simulated Frequency Response and Percentage Error. . . . .	23
13	Time History of Parameter Estimates. . . . .	24
14	Time at Which Parameter Variance Dropped Below the Noise ( $6\sigma$ ). . . . .	25
15	Measured Frequency Response Data from a Woofer. . . . .	26
16	Comparison of Data and Estimate. . . . .	27

# 1 Introduction

This work is motivated by frequency domain system identification of a lightly damped structure with broadband characteristics. Current identification techniques for fitting frequency domain data often fall short of identifying such systems [2, 6, 14]. These techniques typically model the plant as a rational function, in which the order of the numerator and denominator polynomials is chosen to agree with the (presumed) order of the unknown system. The numerical problems associated with high-order polynomials imposes a ceiling on the size and complexity of the problems that can be solved. As described in [1], this difficulty can be overcome by utilizing composite curve fitting, in which two (or more) models are identified in parallel, with each model fitting the system dynamics accurately over a fixed frequency band.

This report supplements a paper submitted by Fisher, Jue, Packard and Poolla to the American Control Conference [13], providing additional discussion on several of the topics therein. We have developed several Matlab<sup>TM</sup> tools, requiring the  $\mu$  Analysis and Synthesis Toolbox, to facilitate composite curve fitting in the context of continuous-time, multi-input multi-output (MIMO) system identification. These tools provide a minimal, stable state-space realization; in a Controls context, there is no need to retain any of the modal information. Although this methodology is designed for lightly-damped structures, we propose that it is valid for the entire class of stable systems.

The remainder of this paper is organized as follows: Frequency domain techniques are discussed in Section 2, including the Santhanam-Koerner [19] curve fitting algorithm and Bayard's multi-band fitting procedure. Additionally, computational details of treating MIMO plants using these methods are discussed in Section 2. Section 3 discusses how we might obtain the transfer function data. Several examples are discussed in Section 4.

Notationally, the  $m$ -input,  $p$ -output, continuous-time, linear time-invariant system to be estimated will be written  $H$ . The frequency response of  $H$  at frequency  $\omega_k$  will be written  $H_k$ . Frequency response *data* will be denoted  $g_k^d$  for the scalar case, and  $G_k^d$  for the vector and matrix cases. For a matrix  $M \in \mathbf{C}^{p \times m}$ ,  $\text{vec}(M) \in \mathbf{C}^{pm}$  denotes the vector formed by stacking its columns. Euclidean norms for vectors and Frobenius norms for matrices are used.

Matlab<sup>TM</sup> is a registered trademark of The Math Works, Inc.  $\mu$ -tools is a product of MUSYN Inc.

## 2 Frequency Domain System Identification

To facilitate the discussion of this system identification method, we will first look at the solution for a single input-single output, SISO, system and then adjust the notation and methodology for multi-variable systems.

### 2.1 Fitting Frequency Data with a Rational Function

The rational transfer function for the  $p^{\text{th}}$  order system is the following:

$$H(s) = \frac{n(s)}{d(s)} = \frac{\sum_0^p n_j s^j}{\sum_0^p d_j s^j}, \quad \text{with } d_p = 1,$$

with  $s = j\omega$ , and polynomials of the form  $n(s) = n_0 + n_1 s + \dots + n_{p-1} s^{p-1}$ , and  $d(s) = d_0 + d_1 s + \dots + d_{p-1} s^{p-1} + s^p$ . Note that we choose to write the transfer function so that the coefficient of the highest order denominator term,  $d_p$  has unity value.

From experiments or modeling techniques, frequency response data has been acquired,  $\{g_k^d\}_{k=1}^L$  at  $L$  ordered frequencies  $\{\omega_k\}_{k=1}^L$ . Our goal is to find the coefficients of the transfer function  $\hat{H}(s)$  of pre-specified order that best fits the data. We use a least-squares method to find the "best fit"; thus we pose the following *nonlinear* optimization problem:

$$\hat{H} = \arg \min_H \sum_{k=1}^L \|q_k(g_k^d - H_k)\|^2. \quad (1)$$

where  $H$  is the  $p^{\text{th}}$  order, strictly proper transfer function indicated above.

The relative importance of individual frequencies is expressed by a weighting curve,  $q_k$ .

## 2.2 Santhanum-Koerner Algorithm

The nonlinear optimization problem (1) can be effectively attacked using the Santhanam-Koerner (SK) algorithm [19]. By casting the problem into a *linear* least-squares form, iterative least-squares methods can be used. The optimization problem (1) can now be expressed:

$$\hat{H} = \arg \min_H \sum_{k=1}^L \left\| q_k \left( g_k^d - \frac{n(s)}{d(s)} \right) \right\|^2 = \arg \min_H \sum_{k=1}^L \left\| \frac{q_k (g_k^d d(s) - n(s))}{d(s)} \right\|^2.$$

The SK algorithm involves locating a starting point (an initial guess), and then iterating the problem until a solution, or a sufficiently good solution, is obtained. Thus we begin the iteration with  $i = 0$  and an initial estimate  $n(s)^0, d(s)^0$ . This initial estimate involves making a guess as to the model order required to fit the data. Let's say we assume a model order of  $p$ , thus we start the iteration with  $d(s)^0 = (s + \gamma)^p$ ; our initial guess for the numerator follows  $n(s)^0$  easily. Subsequent estimates are obtained by using the previous estimate for  $d(s)$  and solving:

$$(n(s)^{i+1}, d(s)^{i+1}) = \arg \min_{n,d} \left\| \frac{q_k (g^d d(s)^{i+1} - n(s)^{i+1})}{d(s)^i} \right\| = \frac{1}{d(s)^i} \left( \arg \min_{n,d} \|q_k (g^d d(s)^{i+1} - n(s)^{i+1})\| \right). \quad (2)$$

Thus, for the  $(i + 1)$ th iteration,  $d(s)^i$  is known, and the  $n_j$  and  $d_j$  parameters enter the problem linearly. If  $n(s)$  and  $d(s)$  are parameterized using linear basis functions expansions (see for example [21]), this is an attractive (weighted) least-squares problem.

### 2.2.1 Bilinear Transform

The SK procedure is subject to numerical conditioning problems associated with basis function choice. For continuous-time problems, the associated least-squares problems involve Vandermonde matrices containing large powers of  $s$ , and thus large powers of  $\omega_k$ . This difficulty can be handled by using a bilinear transformation to map the frequencies  $\{\omega_k\}$  onto the unit circle, and solving the resulting discrete-time problem. For  $\alpha > 0$ :

$$\hat{z}(\alpha, s) = \frac{s - \alpha}{s + \alpha}.$$

Note that for  $s = j\alpha$ ,  $z = e^{j\pi/2}$ . Furthermore, for  $s = j\omega \iff \|z\| = 1$ . Data near the origin now lies near -1 on the unit disk, and data near  $\infty$  lies near 1 on the unit disk. This transformation provides us with a mapping of a rational function in  $s$  to a rational function in  $z$ , and the inverse transformation, back to  $s$  also maintains rationality. The imaginary axis is mapped onto the unit circle, and poles in right-half s-plane are mapped to the outside of the unit circle,  $\|z\| > 1$  and poles in the left-half of the s-plan are mapped to the inside of the unit circle,  $\|z\| < 1$ .

In fact, since we are assuming a stable system, we expect that all of the data will be mapped onto or within the unit disk; we can use powers of  $e^{j\omega_k}$ , which are periodic and present less numerical conditioning problems. The resulting model is then re-transformed onto a continuous-time model by inverting the bilinear transformation. This procedure preserves rationality of transfer function matrix models.

The bilinear transformation also provides another means of weighting the data. The analog frequencies are "warped" through the bilinear transformation; the left and right ends of the frequency range are compressed while those frequencies in the middle are spread out on the top of the unit disk. The result of this warping is that the frequency points which are mapped to the top of the unit disk are easily fit with a low order model in the least-squares sense, while those approaching  $|z| = 1$  require a much higher order to fit; however, a very high model order estimate may be unstable, in which case we discard those estimates. Thus, the data along the top of the unit disk will be fit better and earlier than the data at the extremes. We can select any continuous frequency  $\alpha$  and have it mapped to the discrete frequency  $\pi/2$ . This flexibility permits us to "pre-warp" the data so that the continuous frequency range of interest (centered around  $\alpha$ ) is favorably placed on the unit circle. This technique is often used in digital filter design to retain the analog filter transition bands in the digital filter equivalent. The choice of the "pre-warp" frequency is left to the modeler, but a reasonable choice is the geometric mean of the minimum and maximum frequencies, (i.e.  $\alpha = \sqrt{\omega_1 \omega_L}$ ).

As an example, if there is a significant amount of “flat” response data (e.g., at low frequencies, or between peaks), it will be helpful to choose a different pre-warp term than one defined by the high and low ends of the data range. The lack of variability in this “flat” data will cause difficulties in the least-squares problem; the set of equations will be indeterminate due to loss of rank. In this case, it may be helpful to choose  $\alpha = \sqrt{\omega_a \omega_b}$ , where  $\omega_a$  and  $\omega_b$  correspond to the endpoints of the range in which there appears to be more “action”. Another method would be to use the most resonant peak as the pre-warp term:

$$\alpha = \arg \max_{\omega} (g^d(\omega)).$$

The quality, stability and numerical validity of the estimate can be somewhat sensitive to the choice of  $\alpha$ . It appears that for data with broad spectral characteristics, the use of the geometric mean provides reasonable results; for regions of data that are highly resonant, fine tuning this value, even to within a few radians along the unit circle, can improve the stability of an estimate for a given model order. With the speed of computers available today, it could be possible to optimize this term automatically.

### 2.2.2 The Least-Squares Equation

We will show the least-squares derivation in terms of powers of  $z$ . Say we have selected a model order,  $M$ , so that our desired discrete polynomials will have the form:

$$d(z) = 1 + d_1 z^{-1} + \dots + d_M z^{-M}$$

and

$$n(z) = n_0 + n_1 z^{-1} + \dots + n_M z^{-M}.$$

The minimization problem (2) now appears as the following in the  $Z$ -domain

$$\min_{n, d} \left\| \frac{q_k}{d^i(z)} [g_k^d d^{i+1}(z) - n^{i+1}(z)] \right\| = \min_{n, d} \sum_{k=1}^L \left\| \frac{q_k}{d^i(e^{j\theta_k})} \underbrace{[g_k^d(e^{j\theta_k}) d^{i+1}(e^{j\theta_k}) - n^{i+1}(e^{j\theta_k})]} \right\|^2,$$

where the bracketed terms on the right side can be rewritten to form a vector of the known parameters,  $Y$ , a vector composed of the scaled powers of  $e^{j\theta_k}$ ,  $\phi$ , and a vector of the unknown parameters,  $\Theta$ :

$$g_k^d(e^{j\theta_k}) [1 + d_1 e^{-j\theta_k} + d_2 e^{-j2\theta_k} + \dots + d_M e^{-jM\theta_k}] - [n_0 + n_1 e^{-j\theta_k} + \dots + n_M e^{-jM\theta_k}]$$

$$= g_k^d(e^{j\theta_k}) - [\phi(e^{j\theta_k})] \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_M \\ n_0 \\ \vdots \\ n_M \end{bmatrix} = Y - \phi\Theta.$$

This is the weighted least-squares equation of the SK algorithm, where everything has now been weighted by the previous estimate of  $d(z)$ :

$$\min_{\theta} \|Y - \phi\Theta\| = \min_{n, d} \sum_{k=1}^L \left\| \frac{g^d(e^{j\theta_k})}{d^i(e^{j\theta_k})} - \frac{\phi(e^{j\theta_k})}{d^i(e^{j\theta_k})} \Theta^{i+1} \right\|^2.$$

### 2.2.3 Further Comments on the SK Algorithm

Unfortunately, it has been shown that the SK algorithm is not globally convergent. In particular, the resulting model may be *unstable*. On the other hand, extensive experimental evidence suggests that it performs remarkably well. However, for very high order systems, especially for lightly damped frequency responses, the SK algorithm can yield inadequate or unstable results.

In our experience, the actual number of SK iterations required to achieve a “reasonable” fit may vary from 1 to 5 or as many as 20 or more. We have observed that fitting “ideal” data with the exact number of states results in an oscillating iteration (in the 2-norm sense) in the least-squares; that is, while the error may be very small, it does not (absolutely) converge. However, fitting “real” data (i.e., noisy) with the exact number of states does result in a converging iteration. We use this fact in determining the number of SK iterations and automatically stop iterating once the relative cost (in the 2-norm sense) falls below a user-specified tolerance.

The SK algorithm applies  $1/|d(s)^i|$  as the weight in (2) to solve for  $n(s)^{i+1}$  and  $d(s)^{i+1}$ , and we have found that weighting by  $1/|\sqrt{d(s)^i}|$  helps us obtain a more uniform fit, as discussed in [18]. That is, using the *square root* of the previous denominator, we can de-emphasize the lightly damped modes during the next iteration. This comes at the cost of an increased error, however, a higher model order can thus be used, and we find that ultimately a better estimate is obtained with  $1/|\sqrt{d(s)^i}|$ .

## 2.3 Multi-Band Fit

Transforming the data to the unit disk is not enough to overcome the numerical problems for very high-order, lightly damped systems. The need for a system identification tool for such systems has motivated the multi-band fitting methods of Bayard [1, 4]. The essential idea is to divide the frequencies of interest into sub-bands. Models that fit the data accurately on each band are determined using the SK algorithm, and these are combined to produce a model for the overall system. As each model is fit, we take into account that which is leftover from fitting the other models. We are only concerned with fitting a stable system with a stable model. While it is possible to estimate the total system with a stable model comprised of the sum of non-stable systems, there are potential physical behavior differences between the stable and non-stable systems which we do not wish to encounter.

Our least-squares algorithm takes advantage of the fact that we are fitting data from one dynamical system, and our weighting function is the same for all output channels. Sparse matrix techniques (orthogonal triangular decomposition, QR) have improved the numerical stability and speed of our implementation of the SK procedure; it can fit data with at least 50 or 90 states, depending on the data. This is described more precisely below.

We’ll start by restricting our attention to two bands in solving the optimization problem (1). Define complimentary weighting functions  $q^\ell$  and  $q^h$  by

$$q_k^\ell = \begin{cases} 1 & \text{for } \omega \leq \omega_c \\ 0 & \text{else} \end{cases}$$

and  $q^h = 1 - q^\ell$ . Initialize the estimate with  $q^h = 0$  and  $\hat{H}^h = 0$ . Determine a good model  $\hat{H}^\ell$  in the low-frequency band using the SK algorithm:

$$\hat{H}^\ell = \arg \min_H \|q^\ell q(g^d - (H + \hat{H}^h))\|.$$

Next, determine a good high-frequency model by fitting the *residuals* using the SK algorithm:

$$\hat{H}^h = \arg \min_H \|q^h q(g^d - \hat{H} - H^\ell)\|.$$

Iterating between these two steps will produce a model  $(\hat{H}^\ell + \hat{H}^h)$  that serves to match the wide-band frequency response data. This procedure circumvents the numerical problems associated with the SK procedure and high-order systems mentioned above.

As pointed out in [1], the use of *indicator* weighting functions  $q^\ell$  and  $q^h$  (2.3) is to be avoided because the DC bias is incorrectly identified by this procedure. Bayard suggests a modification that addresses this problem and also improves the “dove-tailing” of the composite model in the low-frequency band.

We have successfully used three tiered weights. For the frequency band  $\Omega_b$ , the weight we employ is

$$q_k^b = \begin{cases} \mathcal{Q}_\ell & \text{if } \omega_k < \omega, \text{ for all } \omega \in \Omega_b \\ \mathcal{Q}_h & \text{if } \omega_k > \omega, \text{ for all } \omega \in \Omega_b \\ 1 & \text{else} \end{cases} \quad (3)$$

where  $\mathcal{Q}_\ell$  and  $\mathcal{Q}_h$  are generally less than or equal to 1. With this particular choice of weight, the degree to which the estimate in the current band is improved can be controlled, at the cost of the estimate in the other bands. It should be noted that Bayard’s weight is a special case of this in which  $\mathcal{Q}_\ell = 1$  and  $\mathcal{Q}_h = 0$ . We have found that choosing  $\mathcal{Q}_\ell \approx 10^{-1}$  and  $\mathcal{Q}_h \approx 10^{-4}$  typically results in better overall performance. The entire procedure generally converges much sooner and requires far fewer SK iterations to estimate each band than with indicator weights, or no weights.

Since the choice of  $\mathcal{Q}_\ell$  and  $\mathcal{Q}_h$  will often depend on the nature of the problem being solved, we add these parameters along with model order  $m$  and  $\alpha$  to the list of parameters that the modeler can select when fitting each band.

### 2.3.1 Dividing the Frequency Range into Bands

It is important to understand that Bayard’s method of dividing the *frequency data* into bands does not require filtering, nor does it introduce model order to the estimate. These are purely mathematical/data analysis constructs which do not affect the physical meaning of the resulting estimate.

It may also seem intuitive to divide the frequency data into as many bands as possible, obtaining a model for each peak. However, note that each band introduces a residual, which every other band will try to accommodate. With  $n$  bands, there will be  $n(n - 1)$  passes at fitting the residual. Unless the bands are selected with such intuition and accuracy so as to enable the user to pick the model order precisely with small error, there will be significant residual dynamics which will manifest in a higher than desired model order required to fit the data.

Instead, the best route to take will involve as few bands as possible, though it may require several iterations to pick the optimal cutoff frequencies the first time around.

### 2.3.2 Recommended Fitting Technique

In iterating between 2, 3, ...,  $n$  bands, there are an infinite number of paths to arrive at the optimal system represented by fixed model orders for each band. For the two band case, the path is easy: optimize the low frequency model first by selecting a model order which minimizes the norm of the error; then move to the high frequency model and similarly find the best model order. Iterate.

When we return to the first model, we may find that it is no longer stable. Thus, a certain amount of balancing and proper band definition between the two models is required to ensure two stable systems. Once we are assured that we have two stable systems, we iterate through the models until the error converges to a user-specified tolerance.

For more than 2 bands, we propose that the proper technique involves fitting the lowest band, then the second band, then iterating between the first two bands until we reach some moderate convergence. Only then do we start fitting the third band; thus the two lower bands are treated as the “lower band” and the third band is the “higher band”. Again, some corrections to the model order may be required as we seek the balance between the  $n$  bands.

In fact, the best fitting algorithm appears to be the following:

- Fit data with one band, optimizing the pre-warp term and model order to minimize the error norm; this value should be the maximum error we can possibly obtain with multi-band methods. The resulting estimate with just one band may be the best estimate possible.
- Divide the data into two frequency bands. Choose the cut-off frequency at an area of “zero slope”. Preferably, this cut-off frequency will lie in the left-half of the frequency range.
- Find an optimal pre-warp term and model order for the low frequency data. Pay more attention to error norm

of this band, rather than the total error. The ideal model for this band will drop off or remain constant through the upper frequency data. It may be possible to fit a few peaks in the high frequency band with this  $\hat{H}^L$ .

- Find an optimal pre-warp term and model order for the high frequency data. Again, focus attention on the error norm of this band. The ideal model for this band will drop off through the lower frequency data. It will probably be possible to fit the higher amplitude peaks in the low frequency residual with  $\hat{H}^H$ .
- Iterate several times over these models. Ideally, the total error norm will drop with each complete iteration. If not, modify the model order of each  $\hat{H}^i$ , or adjust the location of the cut-off frequency. The error norm may worsen slightly between the low and high frequency estimates, but the total error norm should improve after the high band has been fit. It may be possible that a third band is required, in which case, choose another “flat” area in the high frequency, or larger frequency range band, and repeat this process for the lower two bands.
- For more than three bands, once the first the estimates (sub-models) for the first two bands start converging, it is appropriate to fit the next band, until the lower three models converge, and so on.
- Once the model orders have been selected, it can be useful to start again from scratch, using the final model orders and pre-warp values selected, and repeating the previous steps. It should be possible just to set the model orders and pre-warp for the first two bands and iterate (at least three times) and then add the third model, and so on. The estimate should converge more rapidly and to a lower number than was obtained previously.

It is the modeler’s decision to stop the iteration process. If all the parameters have been chosen correctly, this process should produce a result superior to the one-band fit.

This heuristic approach can be extremely time consuming, particularly for large numbers of inputs, outputs and states. It will be desirable to find an analytical means of selecting the cut-off bands and pre-warp terms.

## 2.4 Multi-Variable Systems

### 2.4.1 Vector case (SIMO)

The previous equations are easily adapted for a multi-output system, as the desired rational function is merely composed of a vector of numerator functions,  $N(s)$ , and (2) now becomes:

$$(N^{i+1}(s), d^{i+1}(s)) = \arg \min_{N,d} \left\| \frac{Q(G^d d(s) - N^{i+1}(s))}{d^i(s)} \right\|. \quad (4)$$

Additionally, note that the weighting curve is now a matrix as well. It is straightforward to incorporate left and/or right weights as needed. MISO frequency data can easily be transposed to a SIMO form and fit as mentioned above, however the resulting system will need to be reshaped to the proper MISO dimensions. This procedure is the same as that discussed below for the full MIMO case.

### 2.4.2 Matrix Case (MIMO)

Given that all of the columns of the data were generated with the same system, we expect the underlying denominator (defined by the  $A$  matrix) to be the same for all of the columns (inputs). However, if we fit each column separately, there is the possibility that the full denominator is not identifiable for a given (arbitrary) column of data. Furthermore, due to the noise and uncertainty of our experimental data, we may obtain a system estimate with eigenvalues for each column which vary in a significant numerical way, even though they are in fact the same. There are two methods to “fix” the denominator for each column: 1) perform a column fit, say, on column 1 (1st input) data, and keep that  $A$  matrix for the subsequent columns, or 2) reshape the MIMO data to a SIMO data set, thus fitting all of the columns of data simultaneously.

The principal advantage of the first method is that the overall system structure is maintained (MIMO). From a computational standpoint one can focus on a subset of the total data set, and perhaps the cut-off bands and model orders for the first (fitted) column may be set and one could perform an auto-fit on the remaining columns. The disadvantage is there is no way to know ahead of time which column to fit first. Furthermore, what if no one column really contains enough data to identify the full  $A$  matrix?

The principal advantage of the second method is that all of the data is fit at once with a SIMO system; all of the data is considered in identifying the underlying  $A$  matrix. Furthermore, computationally, there is no need to carry over the  $A$  matrix, cut-off bands or model orders to another sub-fit routine. The process of choosing the cut-off bands for all of the data is simplified, since all of the data is visible; there is less chance of inadvertently picking a cut-off frequency which works well for some columns of data but not for others. (Of course, there may be significant disadvantages to this method due to wildly differing resonances at each input.) The main disadvantage of this path is that the resulting system matrix must be re-shaped to the original MIMO configuration.

We have implemented the second method, and in the next section, the re-shaping of the fitted system from a (long) SIMO system to the correct MIMO system is discussed.

### 2.4.3 Realization of a MIMO System

Restoring the SIMO system estimate to the proper MIMO shape is not a trivial task. Our method uses the so-called Gilbert realization.

Our system identification algorithm requires a SIMO (or SISO) system, hence it returns to us a SIMO (SISO) system which needs to be reshaped to the original MIMO dimensions through manipulation of the  $b$ ,  $C$  and  $D$  matrices.

From Gilbert, the transfer function can be expressed as a partial fraction expansion if we assume simple poles. This realization can be written as a matrix of the poles multiplied by the product of the “B” and “C” matrices,  $M$ .

$$H(\omega) = \sum_{i=1}^k \frac{1}{s - \lambda_i} M_i. \quad (5)$$

where  $M_i$  is a  $p \times m$  matrix, and  $M_i = L_i R_i$ , where  $L_i$  is  $p \times r_i$  and  $R_i$  is  $r_i \times m$ , and  $r_i$  is the rank of  $M_i$ . Thus, a minimal realization of our system can be expressed as the system matrix:

$$\left[ \begin{array}{ccc|c} \lambda_1 I_{r_1} & & 0 & R_1 \\ & \lambda_2 I_{r_2} & & R_2 \\ & & \ddots & \vdots \\ 0 & & & \lambda_k I_{r_k} & R_k \\ \hline L_1 & L_2 & \cdots & L_k & 0 \end{array} \right].$$

The output of our system identification tool is a system matrix in this form, where the  $A$  matrix is expressed in a “bi-diagonal” or block diagonal form (real-valued), where  $\lambda = \sigma + i\omega$ :

$$A = \left[ \begin{array}{cccc} \ddots & & & \\ & \sigma & -\omega & \\ & \omega & \sigma & \\ & & & \ddots \end{array} \right].$$

In order to sort and reshape this system, we need to convert  $A$  to a strictly diagonal matrix (complex-valued), sort the eigenvalues and perform a similarity transformation on the  $A$ ,  $b$  and  $C$  matrices. We want to have  $A$  in the form:

$$A = \left[ \begin{array}{ccc} \ddots & & \\ & \lambda & 0 \\ & 0 & \bar{\lambda} \\ & & & \ddots \end{array} \right].$$

Thus, we require a transformation of the SIMO system representation with bi-diagonal eigenvalues to a SIMO system with complex conjugate pairs.

Suppose the true system is  $p \times m$ , with  $n$  states. We have stacked our frequency data to get a SIMO system, and obtain an  $A, b, C$  for the SIMO system.

First we need to diagonalize  $A$ . We assume that our system estimate gives us discrete, not repeated, poles. Thus, the eigenvalue matrix  $T$  has full rank and  $A$  is diagonalizable, and the following will retain full row rank:  $[T, \Lambda] = \mathbf{eig}(A)$ . Then  $\tilde{b}$  has only one column, and  $\tilde{b}$  and  $\tilde{C}$  are complex valued, since  $T$  is complex valued.

Next we sort the eigenvalues by the real values, and then sort by the absolute value of the imaginary value. This will ensure that the real-valued eigenvalues are kept together, and that the conjugate pairs are grouped together. Keep track of the sorted array indices and sort the columns of  $T$  accordingly, to find the re-ordered eigenvalue matrix  $\tilde{T}$ . Use  $\tilde{T}$  to transform the system matrices to an ordered system representation. Thus

$$\begin{aligned}\tilde{A} &= \tilde{T}^{-1}A\tilde{T} = \Lambda_{\text{sorted}} \\ \tilde{b} &= \tilde{T}^{-1}b \\ \tilde{C} &= C\tilde{T}.\end{aligned}$$

Then calculate  $M_i$  with the  $i$ th column of  $\tilde{C}$  and the  $i$ th row of  $\tilde{b}$  to reshape the system to the original input/output dimensions.  $M_i = \mathbf{reshape}(\tilde{C}_i\tilde{b}_i, p, m)$ .  $M_i$  can then be factored into the product  $L_iR_i$  using a singular value decomposition approach. Recall,  $\mathit{rank}(H_i) = r$ , then

$$\begin{aligned}U\Sigma V &= \mathbf{svd}(M_i) \\ L_i &= U\Sigma_{rc}^{1/2} \\ R_i &= \Sigma_{rr}^{1/2}V^*.\end{aligned}$$

Note that  $\Sigma_{rc} = \Sigma(:, 1:r)$  and  $\Sigma_{rr} = \Sigma(1:r_2, :r)$ . Now we have a complex-valued state-space realization for our system that is  $p \times m$ . Note that:  $L_iR_i = \tilde{C}_i\tilde{b}_i = M_i$  and  $L_{i+1}R_{i+1} = \tilde{C}_{i+1}\tilde{b}_{i+1} = M_{i+1} = \overline{M}_i$ . Thus, where  $i$  is incremented by one for each real-valued  $\lambda_i$ , and by two for each complex-valued  $\lambda_i$ . For each of the  $M_i$  subsystems, one may choose to evaluate the spread of the singular values and reject the singular values  $\sigma_{ik}$  less than  $\Delta\sigma_{i1}$ , where  $\Delta$  is a user specified value. In these cases, the value of  $\mathit{rank}_i$  will be equal to the number of singular values which exceed the tolerance.

Finally, we perform a state transformation to realize our real-valued state-space. This last real transformation is implemented implicitly using (6) and (7) below. We define the transformation matrix,  $F$ , as the following:

$$F = \begin{bmatrix} I_r & jI_r \\ I_r & -jI_r \end{bmatrix},$$

which results in the following equality:

$$F^{-1} \begin{bmatrix} \sigma + j\omega & 0 \\ 0 & \sigma - j\omega \end{bmatrix} F = \begin{bmatrix} \sigma & -\omega \\ \omega & \sigma \end{bmatrix}.$$

The eigenvalues are sorted, so for complex conjugate eigenvalue pairs,  $\lambda_i$  and  $\lambda_{i+1}$ , the corresponding  $L_i, L_{i+1}$  pair and  $R_i, R_{i+1}$  pair are also complex conjugates. Thus

$$\begin{aligned}[L_i \ L_{i+1}] F &= [W + iX \ W - iX] \begin{bmatrix} I_r & jI_r \\ I_r & -jI_r \end{bmatrix} \\ &= [2W \ -2X] = [2\mathit{Re}(L_i) \ -2\mathit{Im}(L_i)],\end{aligned}\tag{6}$$

and

$$F^{-1} \begin{bmatrix} R_i \\ R_{i+1} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} I_r & I_r \\ -jI_r & jI_r \end{bmatrix} \begin{bmatrix} Y + iZ \\ Y - iZ \end{bmatrix} = \begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} \mathit{Re}(R_i) \\ \mathit{Im}(R_i) \end{bmatrix}.\tag{7}$$

For complex-valued eigenvalues we have the following real-valued  $A_i, B_i, C_i$  matrices, where **daug** = diagonally augment, and the index is iterated by two:

$$\begin{aligned}\hat{A}_i &= \text{daug} \left( \hat{A}_{i-1}, \begin{bmatrix} \text{real}(\lambda_i)I_{rank} & -\text{imag}(\lambda_i)I_{rank} \\ \text{imag}(\lambda_i)I_{rank} & \text{real}(\lambda_i)I_{rank} \end{bmatrix} \right) \\ \hat{B}_i &= \begin{bmatrix} \hat{B}_{i-1} \\ \text{real}(R_i) \\ \text{imag}(R_i) \end{bmatrix} \\ \hat{C}_i &= \begin{bmatrix} \hat{C}_{i-1} & 2[\text{real}(L_i) & -\text{imag}(L_i)] \end{bmatrix}.\end{aligned}$$

For the real-valued eigenvalues, the iteration is simplified:

$$\begin{aligned}\hat{A}_i &= \text{daug} \left( \hat{A}_{i-1}, \begin{bmatrix} \lambda_i I_{rank} \end{bmatrix} \right) \\ \hat{B}_i &= \begin{bmatrix} \hat{B}_{i-1} \\ R_i \end{bmatrix} \\ \hat{C}_i &= \begin{bmatrix} \hat{C}_{i-1} & 2L_i \end{bmatrix}.\end{aligned}$$

### 3 Obtaining the Frequency Response Data

Frequency response data can be acquired from simulations or from experiments. In either case, there will be some transient response, effect of initial conditions, and there can be noise.

#### 3.1 Steady State Response and the Transfer Function

How do we know that the steady state response of a stable system excited by a sinusoid gives us the transfer function of the system? What about the transient dynamics? Recall that the convolution integral is the following:

$$y(t) = \int_0^t g(t-\tau)u(\tau)d\tau. \quad (8)$$

If the input is a complex vector at the real frequency  $\omega$ ,  $u(\tau) = \bar{u}e^{j\omega\tau}$ , and  $\eta := t - \tau$  then

$$\begin{aligned}y(t) &= \int_0^t g(t-\tau)\bar{u}e^{j\omega\tau}d\tau \\ &= \int_0^t g(\eta)e^{j\omega(t-\eta)}d\tau \bar{u} \\ &= e^{j\omega t} \int_0^t g(\eta)e^{-j\omega\eta}d\eta \bar{u}.\end{aligned}$$

Furthermore, the integral can be represented as the sum of two infinite integrals, and using the definition of the Laplace transform:

$$\begin{aligned}y(t) &= e^{j\omega t} \left[ \int_0^\infty g(\eta)e^{-j\omega\eta}d\eta - \int_t^\infty g(\eta)e^{-j\omega\eta}d\eta \right] \bar{u} \\ &= G(s)|_{s=j\omega} \bar{u} e^{j\omega t} - e^{j\omega t} \int_t^\infty g(\eta)e^{-j\omega\eta}d\eta \bar{u}.\end{aligned}$$

Thus  $y(t)$  is the sum of the steady state response,  $y_{ss}(t)$ , and the transient response of the system,  $y_t(t)$ . Where the steady state response is the transfer function evaluated at  $j\omega$  times some sinusoidal input.

But how do we know that  $y_t(t)$  truly decays? If we look at the magnitude of  $y_t(t)$  for any  $t \geq 0$

$$\begin{aligned}
 |y_t(t)| &= \left| e^{j\omega t} \int_t^\infty g(\eta) e^{-j\omega\eta} d\eta \bar{u} \right| \\
 &= |e^{j\omega t}| \cdot \left| \int_t^\infty g(\eta) e^{-j\omega\eta} d\eta \right| \cdot |\bar{u}| \\
 &= |\bar{u}| \cdot \left| \int_t^\infty g(\eta) e^{-j\omega\eta} d\eta \right| \\
 &\leq |\bar{u}| \int_t^\infty |g(\eta) e^{-j\omega\eta}| d\eta \\
 &= |\bar{u}| \int_t^\infty |g(\eta)| d\eta.
 \end{aligned}$$

If we assume a BIBO (bounded input, bounded output) convolution system, then  $y_t(t)$  has a finite value, and as  $t \rightarrow \infty$  the value of this integral gets smaller,  $\lim_{t \rightarrow \infty} y_t(t) = 0$ , so that finally the output of the system is

$$y(t) = G(s)|_{s=j\omega} \bar{u} e^{j\omega t} + y_t(t),$$

and the steady-state behavior of the system (as  $t \rightarrow \infty$ ) is

$$y_{ss}(t) = G(j\omega) \bar{u} e^{j\omega t}.$$

Thus we know that the steady state response to a complex sinusoidal input will result in a sinusoidal output, of different magnitude and phase. The relation between the input is our transfer function at  $s = j\omega$ .

## 3.2 Experimental Determination of the Transfer Function

Given a sinusoidal input, we can consider two principal ways to determine the transfer function of any stable system: 1) wait for the "steady state" or 2) perform some calculations on the real-time data and extract some "steady state" information from the transient portion of the signal.

Let's look more closely at the content of the data signal:

$$y(t) = r(t) + A \cos(\omega t) + B \sin(\omega t) + n(t).$$

For a real experiment, we will have effects from initial conditions and the transient,  $r(t)$ , and from noise  $n(t)$ . We assume that our noise has a white distribution.

Since we know the input frequency for each test, and since we will be operating in the linear range of the system, at steady state we can determine the magnitude and phase (or cosine, sine coefficients) rather easily, even in the presence of noise.

### 3.2.1 Waiting For Steady State

The brute force method entails constantly monitoring the output data and comparing the last period or so of data. Given some error criterion,  $\delta$ , we can determine when to stop. For instance we can look at the norm of the error between two adjacent periods of data to determine how close we are to the steady-state. Depending on the noise of the measurement system and the transient characteristics of the physical system under test, this process could conceivably last several minutes per frequency.

We know that the data looks like:

$$y(t) = A^0 \cos(\omega t) + B^0 \sin(\omega t) + n(t) + r(t),$$

where  $A^0$  and  $B^0$  are the true parameters.

Furthermore, given that we are applying a sinusoid of known magnitude and phase, we can correlate the output with the input to obtain an estimate of the magnitude and phase parameters:

$$A(t) = \frac{2}{T} \int_{t-T}^t \cos(\omega_0 \lambda) y^d(\lambda) d\lambda$$

and

$$B(t) = \frac{2}{T} \int_{t-T}^t \sin(\omega_0 \lambda) y^d(\lambda) d\lambda. \quad (9)$$

Once we have achieved steady state, the estimates of  $A$  and  $B$  from (9) are fairly accurate.

### 3.2.2 A Statistical Approach, Finite Time Experiment

But can we use information even while the transient data is not small to determine our parameters? We are motivated to start the parameter estimation process early so that the duration of the experiment is reduced, and so that in using more data, the calculated variance of the measurement noise will be low.

It seems reasonable to find a way to estimate the parameters before, or just as steady state is achieved, but when will that be? Let's say we know what the variance of the noise is.

As the transient settles down, we expect the estimate of the parameters  $A(t)$  and  $B(t)$  to vary on the same order as the noise, or some small multiple thereof (say,  $6\sigma$ ). Furthermore, we expect that at some time  $t_1$ , the estimates of  $A$  and  $B$  will have some small variance, proportional to the noise. We therefore propose that we can get a reasonable estimate of our parameters by averaging the data from  $t_1$  to the final time,  $t_f$ .

$$\hat{A} = \frac{1}{t_f - t_1} \int_{t_1}^{t_f} A(t) dt$$

and

$$\hat{B} = \frac{1}{t_f - t_1} \int_{t_1}^{t_f} B(t) dt.$$

How should we estimate  $t_1$ ?

Let's look at the envelope of the estimates. For instance, for parameter  $A$

$$\begin{aligned} M(t) &= \max[A(\lambda)], \lambda \geq t \\ m(t) &= \min[A(\lambda)], \lambda \geq t \\ \delta(t) &= M(t) - m(t). \end{aligned}$$

And we can estimate  $t_1$  as the intersection of,  $6\sigma$  and  $\delta(t)$ , where  $\sigma$  is the standard deviation, and  $6\sigma$  represents the range in which a large percentage of the of the data lies.

Since we are dealing with a set of sampled frequency points, we have to sum rather than integrate, where  $N$  is the number of data points per frequency,  $N = T/T_s$  and  $p$  is the index of the data point of interest:

$$A(p) = \frac{2T_s}{T} \sum_{k=p-N}^p \cos(\omega_0 k) y^d(k)$$

and

$$B(p) = \frac{2T_s}{T} \sum_{k=p-N}^p \sin(\omega_0 k) y^d(k).$$

## 4 Examples

In this section, we present four examples. The first looks at a simple constructed system, demonstrating the role of the pre-warp term and multi-band fitting. The next example involves system identification of a simulated (virtual) flexible structure using the exact frequency response. For the third example, we've used the finite time sine-wave algorithm to simulate the frequency response for SISO sub-system of the system used in Example 1, with additive and multiplicative noise included for the simulated experiment. The last example illustrates this fitting method using frequency response data from a woofer.

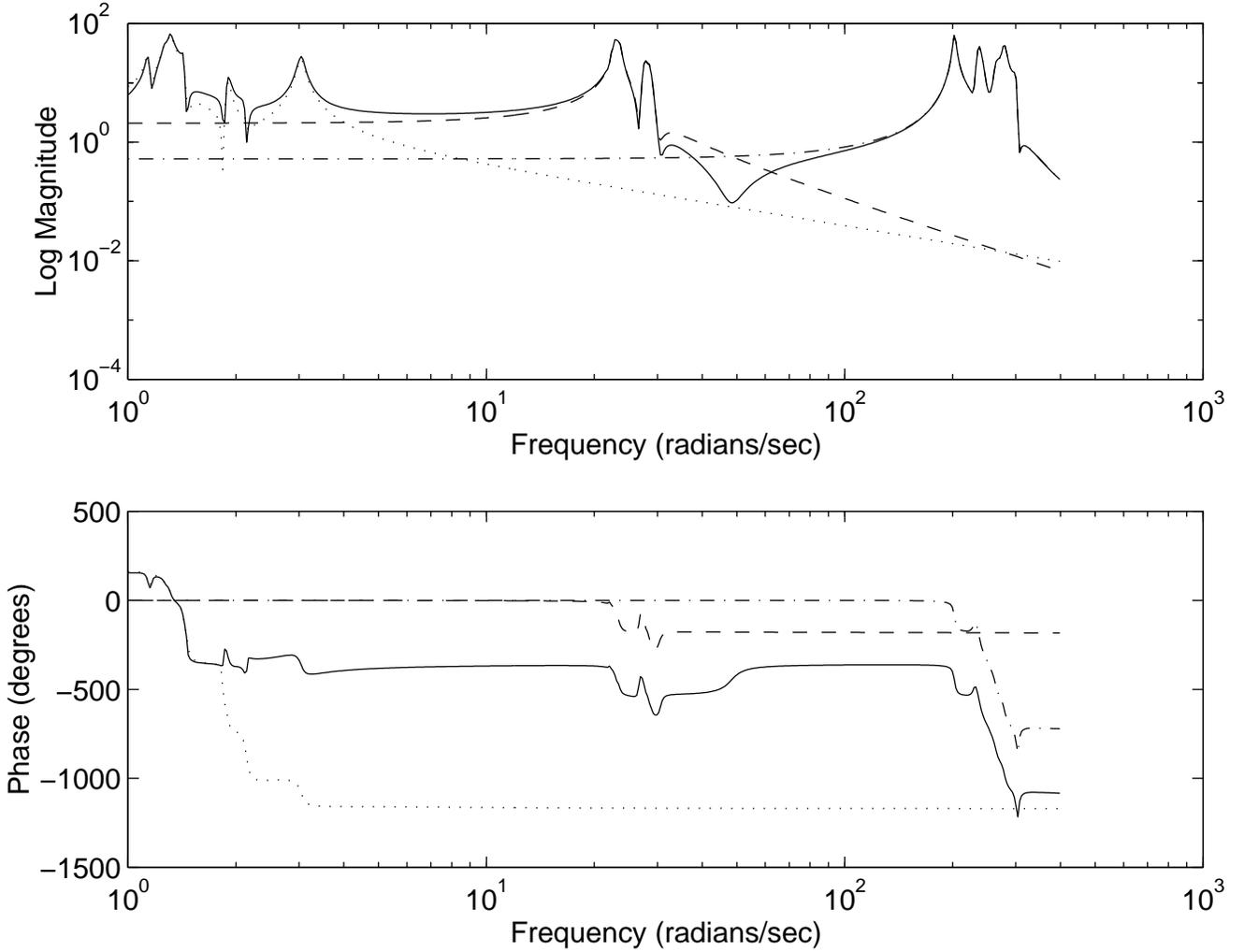


Figure 1: System Composed of Three Separate, Stable Systems.

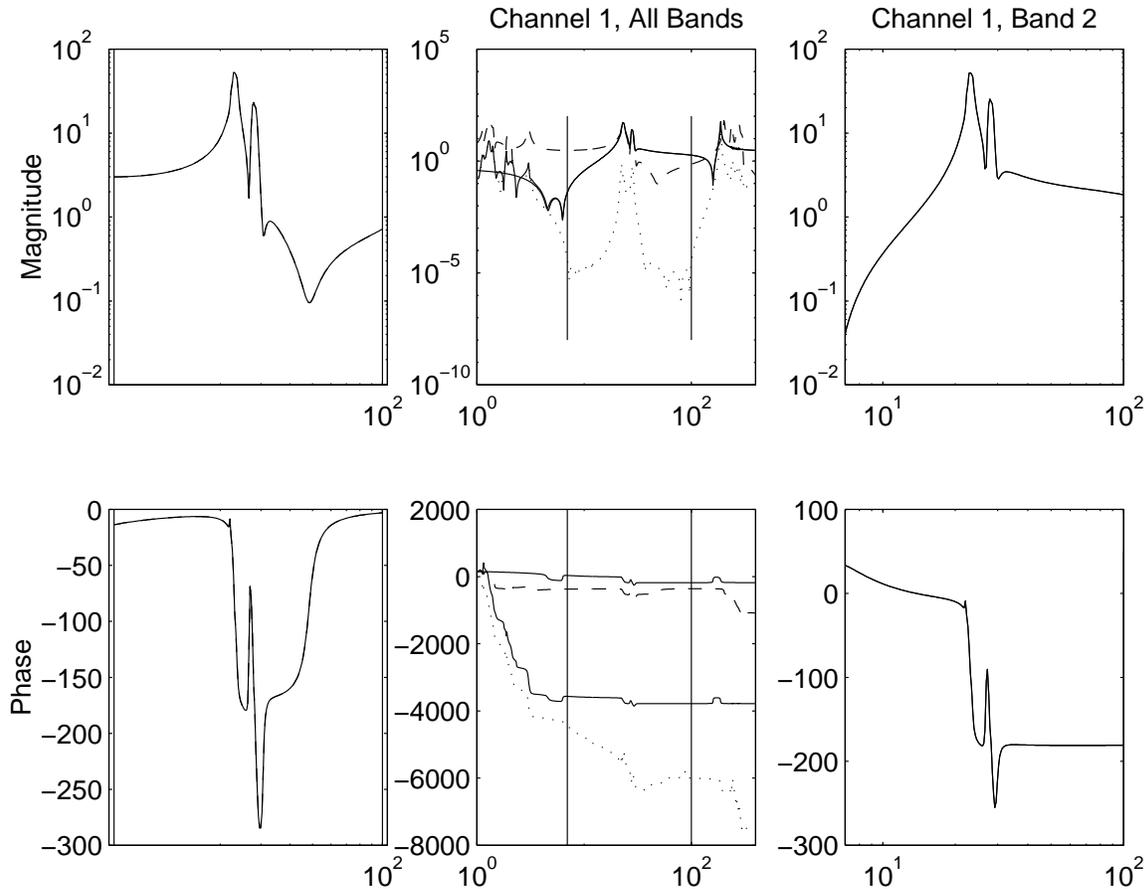


Figure 2: Estimation Display.

#### 4.1 Example 1: Basic Multi-Band Fitting Technique

Figure 1 illustrates the frequency response of a system composed of three systems, each with dynamics in different frequency ranges. The three systems are not quite mutually exclusive, as the low end of the frequency response for the higher frequency systems does not roll-off very quickly. Note that there is some cancellation occurring between the frequencies 30 and 100 radians/sec.

Each system has eighteen resonant modes, arranged as six dominant modes, with a cluster of three resonances at each mode. The separation of the cluster of frequencies was chosen at random within some neighborhood. The damping coefficient for each modal frequency was generated by randomly choosing the damping coefficient from some allowable range. Thus, these systems were generated to exhibit lightly damped behavior.

The estimation tool presents the user with a six-panel figure (Figure 2), in which the first column of panels shows the data (all output channels, if applicable) and the estimate for each band, the second column shows the data for the current (output) channel over the entire frequency range. Also included in this pair of panels is the estimate, residual and total residual. The third column of panels shows the the residual and estimate for the current band and output channel. Since this is a SISO system, only one channel is shown. The resonance frequencies for each system were chosen from a user defined range, and those ranges are indicated by the vertical lines in the central column panels.

Note that the figures in the right column appears to show data from only one system, not the system and the estimate; this is because the estimate, in this case a 20th order system, fit the data in this band very well, with an

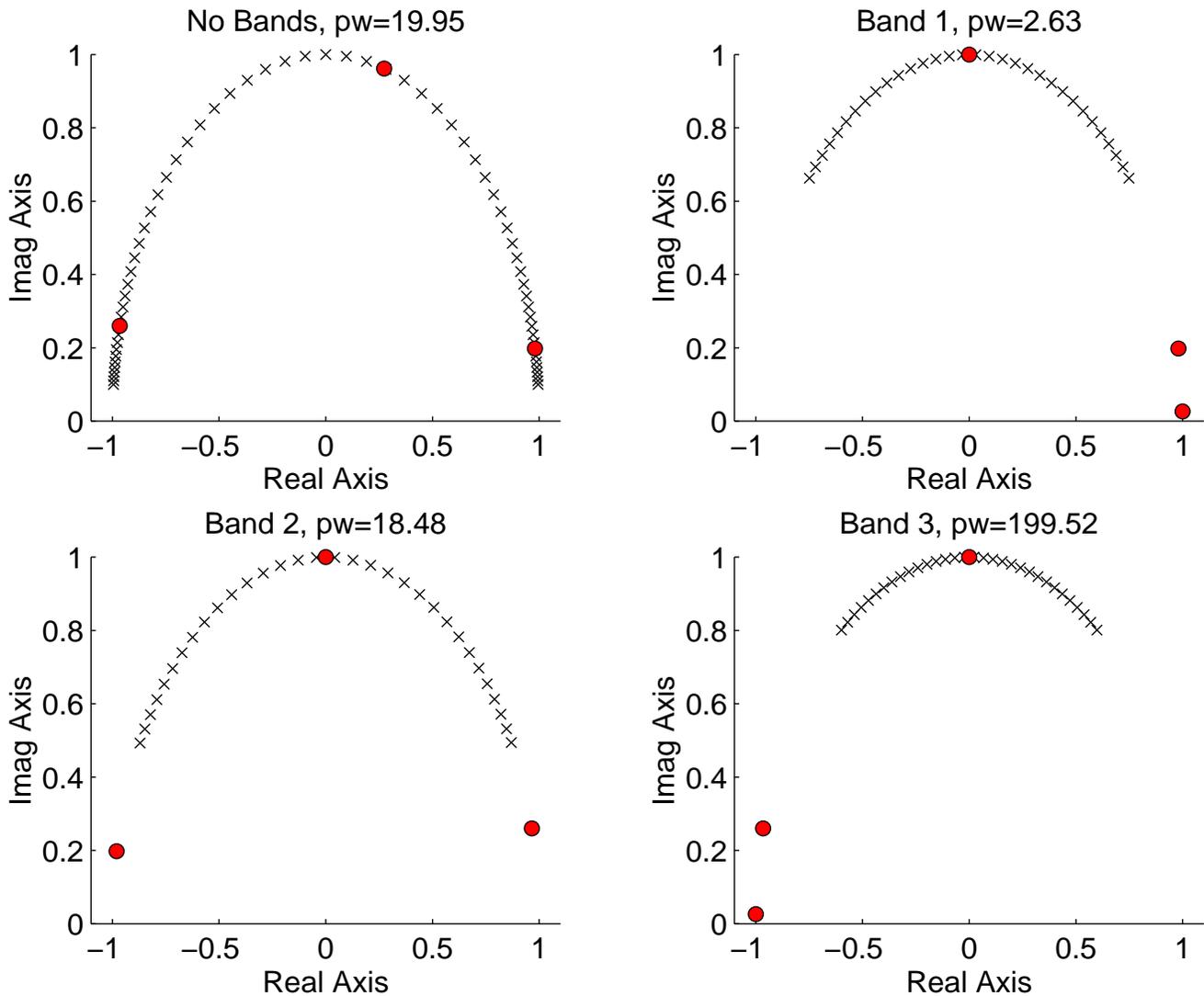


Figure 3: Effect of Pre-Warp Term on Bilinear Transform.

error norm less than 1.6 (normalized 2-norm). The figures in the center column indicate that this system doesn't fit very well outside of Band 2, but that is acceptable, and as we move on to the next Band, our system estimate will fit the *residual* of the estimates for the first two bands, as shown by the dotted line, and not the original system data.

Figure 3 illustrates the effect of the pre-warp term on the bilinear transform. The top left figure shows how the frequency range has been mapped to the unit disk, with the pre-warp term marked by the big dot at  $\pi/2$  radians. The top right figure shows the frequency range in Band 1. See how the frequency ranges, as marked by their respective pre-warp terms, are crowded on the right side of the unit disk, near 0 radians. The lower left figure shows the frequency range of Band 2. In this case, since the frequency ranges for each system were simply scaled by the  $\log_{10}$  of the frequencies, this plot has a very symmetric look. The lower right figure shows the frequency data over Band 3; again, see that the frequency ranges for the lower two bands are crowded on the left side of the unit disk, near  $\pi$  radians.

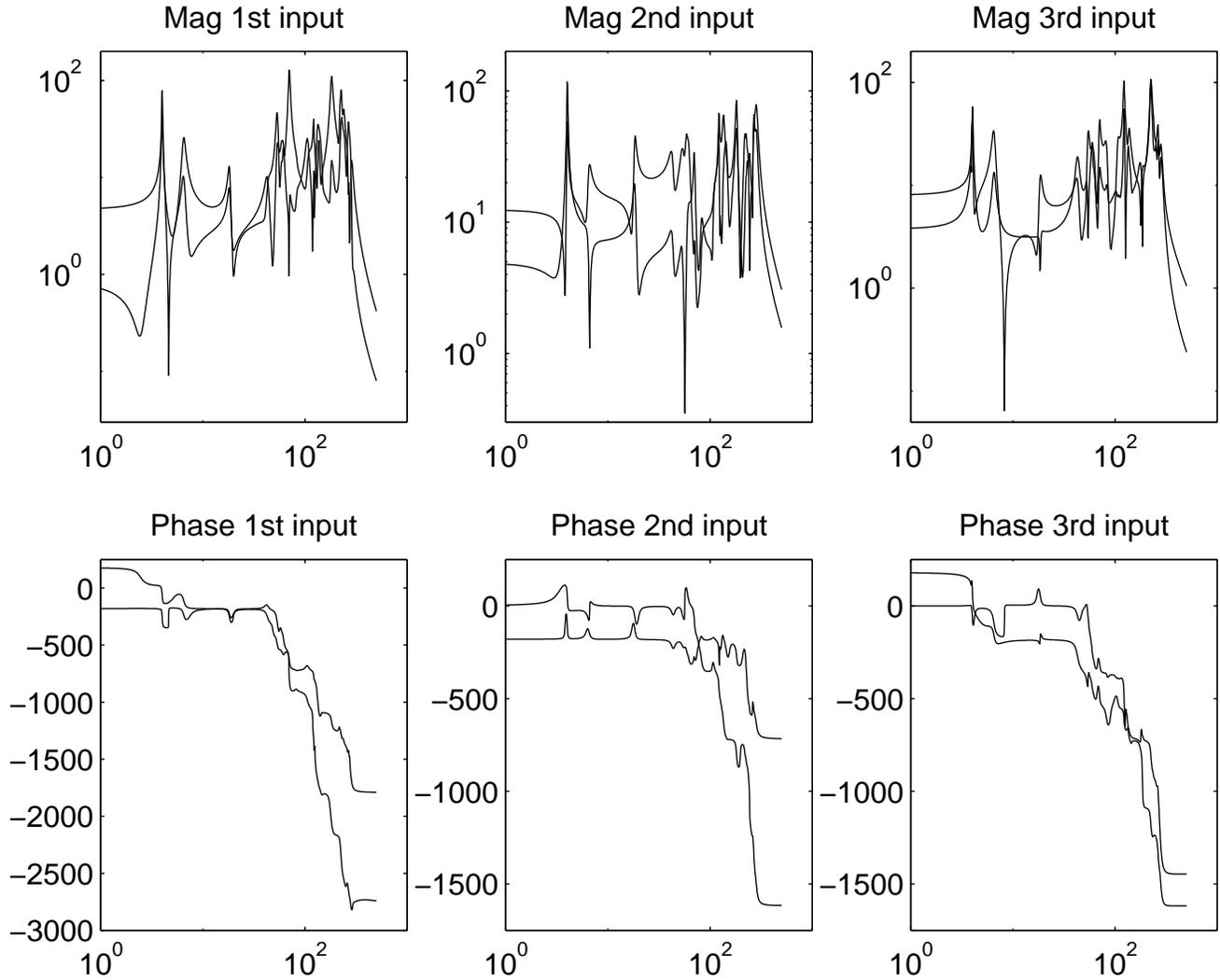


Figure 4: Exact Frequency Response for Virtual System.

## 4.2 Example 2: Fitting Exact Frequency Response Data

We have chosen a small input-output system as our next example. Using a random system generator, we simulated a large-order system with the properties as shown in the Table below.

<i>parameter</i>	<i>value</i>
inputs	3
outputs	2
dominant modes	30
neighboring modes	3
frequency range	1 to 300 rad/sec
frequency separation tolerance	.04
damping coefficient	.01 - .05
total states	180

Table 1: Example 2: Virtual System parameters.

The exact frequency response for all input and output channels for this system is shown in Figure 4. The frequency

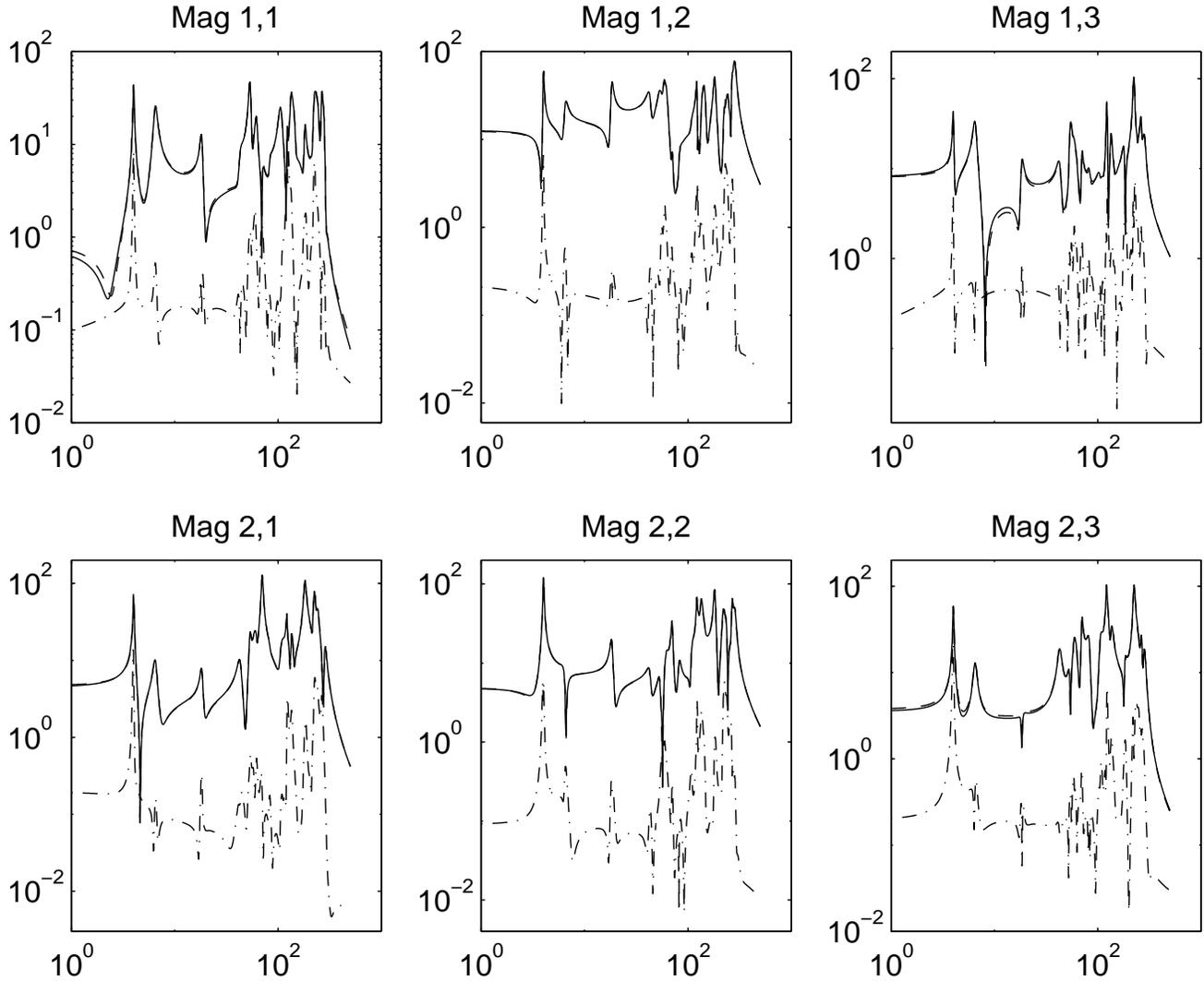


Figure 5: Highest Possible Stable Fit, One Band, 116 states

response for this system was converted to a  $6 \times 1$  system.

With a one-band fit, we were able to find a model with 116 states and a normalized error norm between the system and the estimate of 0.63. This estimate was obtained with a pre-warp term of 127 rad/sec. This estimate is shown in Figure 5. Note that the resonance peaks were not fit very well, with respect to the absolute magnitude.

We can improve on this estimate by using the multi-band method. We chose three frequency bands, with cutoffs at 28.64 and 234.43 rad/sec. The final estimate used a model order of 21 for the low frequencies, 68 for the medium frequencies and 29 for the high frequencies. We optimized the pre-warp term, choosing 10 rad/sec, 87 rad/sec and 245 rad/sec, respectively. Figure 6 shows the difference between the data and an estimate with 236 states, and a normalized error of 0.17. Table 3 lists the progress of the total error of the estimate through 25 full iterations. In comparing these results with those of the single band estimate, note that the low frequency data is fit very well with the multi-band approach. We believe that a proper re-selection of cut-off frequencies in the upper frequencies, and choosing four bands instead of three, would provide an improved estimate.

Figure 7 shows the percentage error between the data and each of the two system estimates presented.

Figure 8 shows ones of windows during the estimation process for this system. Since this is a multi-band system, the far left panels show all 6 outputs of the effective system (SIMO form). The middle and right panels show the data and estimate for the current channel only. We can see that the error in this band (Band 1) is fairly low,

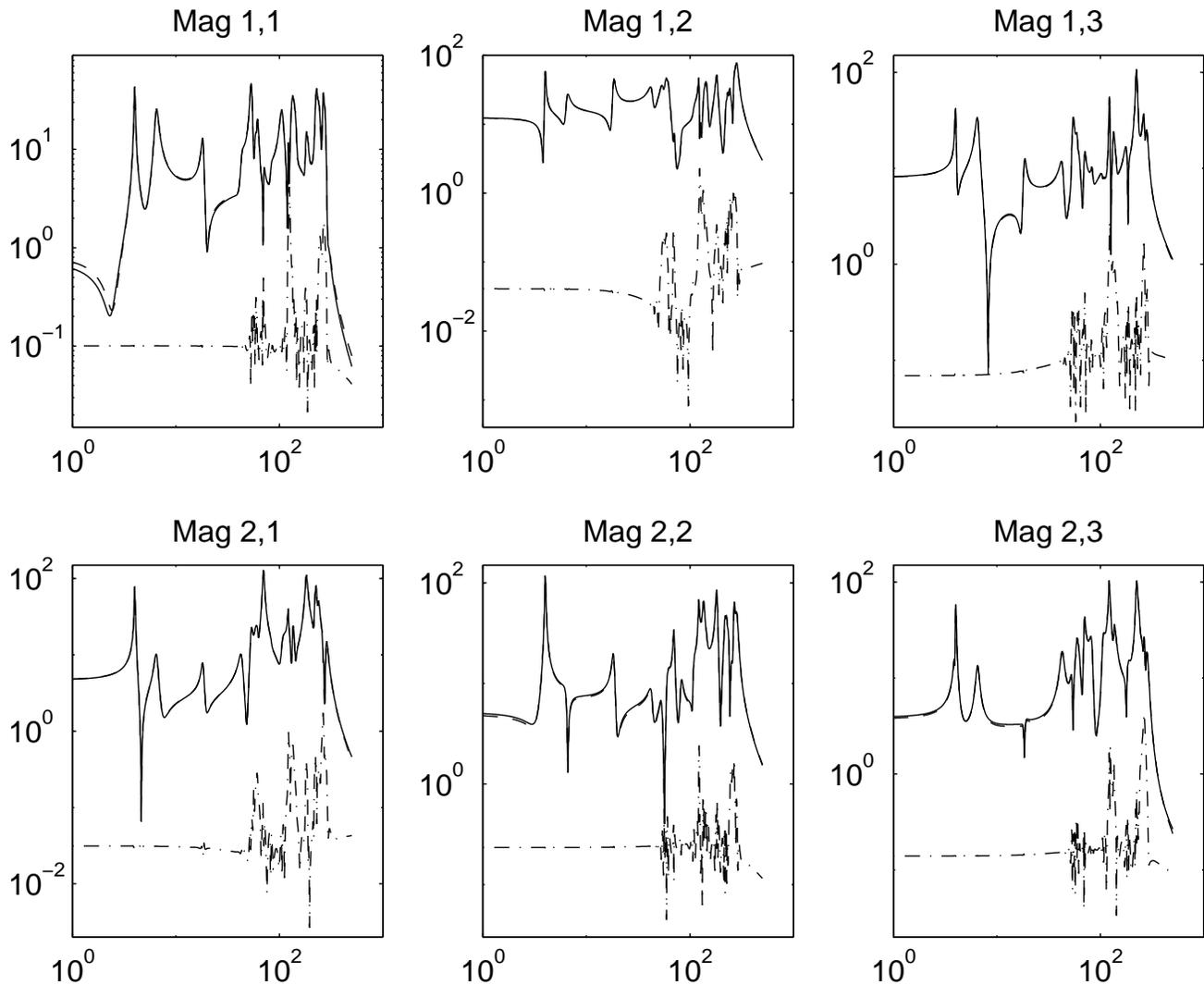


Figure 6: Balanced Multi-Band Solution, Three Bands 236 states

Iter	No. SK iterations			2-norm of Est. error
	band 1	band 2	band 3	
1	3	3	2	0.235
4	2	2	2	0.205
7	3	2	2	0.194
10	3	2	2	0.188
16	3	2	8	0.171
19	3	2	7	0.164
22	3	2	7	0.159
25	3	2	7	0.155

Table 2: Example 2: Iteration summary.

between  $1e-2$  and  $1e-4$ .

Now we have to restore the estimate to the original dimensions, and determine if we can reduce the order of the estimate. Figure 9 shows the Hankel Singular values of the estimated system. If we choose a truncated model order of 230 states, as suggested by the rapid drop off in the singular values, the resulting truncated estimate has an

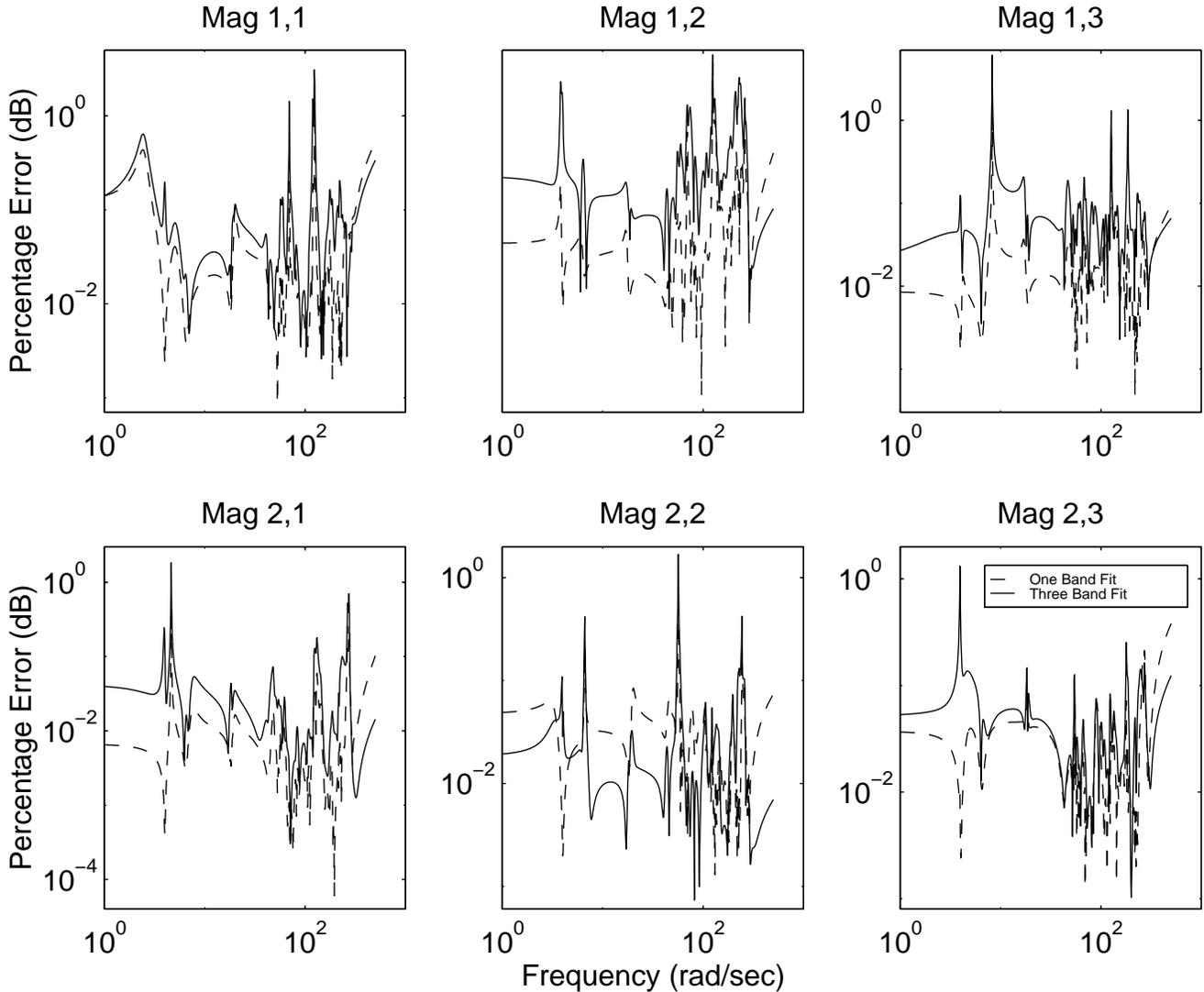


Figure 7: Percentage Error, One Band Fit and Multi-Band Fit.

error of less than  $1e-5$  as compared to the full estimate. We know that the original system has 180 states; choosing a truncated model order of 180 states results in an error of less than  $1e-2$  in all frequencies, compared to the full estimate. (See Figure 10)

Figure 11 compares the poles of the original system and those of the truncated system. We see that a few pairs of poles near the origin are easily determined; the original poles (denoted by  $x$ 's) are spread in a roughly triangular/arrow pattern, while the estimated system places many poles into a set of rectangles, roughly bounded by  $\pm j150$  and  $\pm j250$ . The estimated system also has seven pairs of poles beyond this figure, located between  $(-20, \pm j250)$  and  $(-1400, \pm j750)$ . Truncating this estimate further, to 173 states, moves some of these poles towards the origin, several real poles are introduced.

### 4.3 Example 3: Fitting Simulated-Experimental Frequency Response Data

The simulated data shown in Figure 12 was obtained using the algorithm discussed in Section 3.2.2. Additive noise of .1% and multiplicative noise of .001% were applied to the simulation.

Figure 13 shows the time varying parameter estimates for A and B at a frequency of 2 rad/sec. The lower figures detail the range of the estimate envelope for each parameter and  $t_1$  is defined at the point where this curve crosses

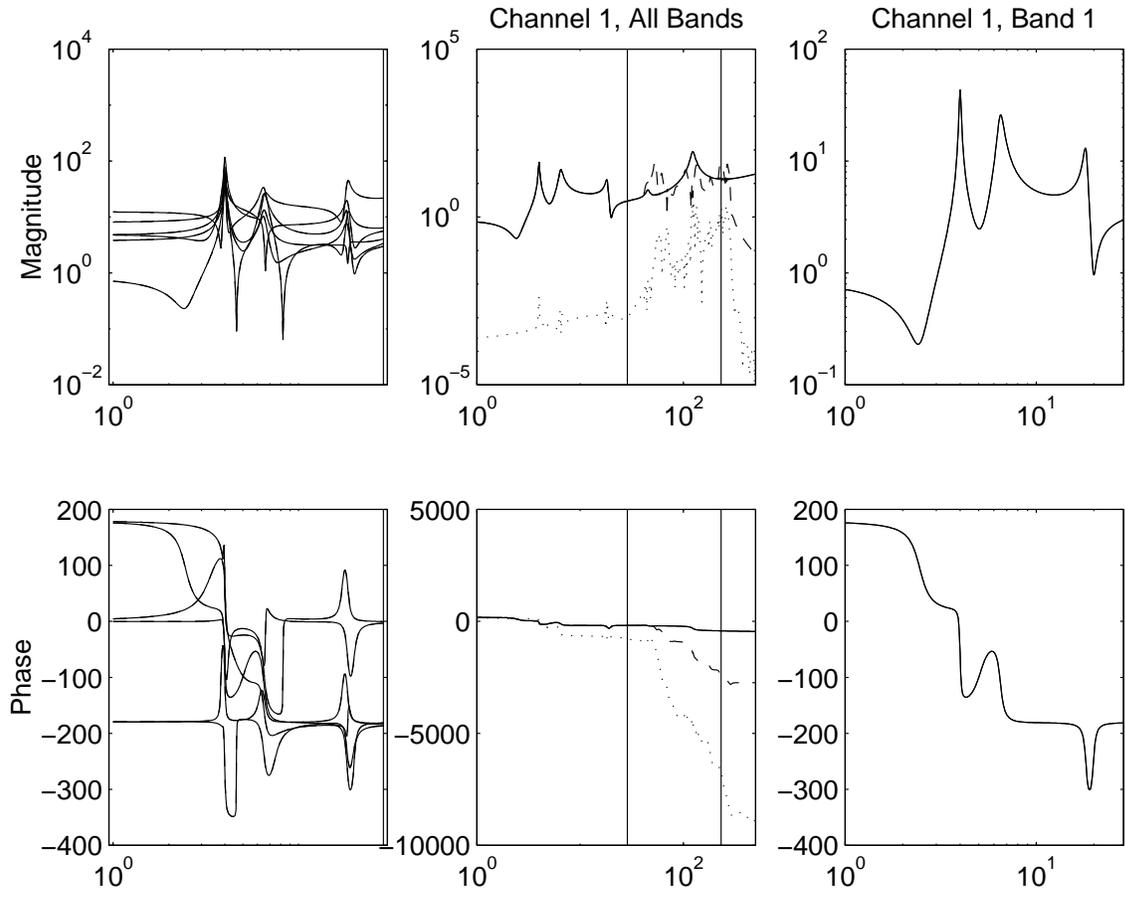


Figure 8: Display During Estimation Process.

the threshold.

Figure 14 illustrates the estimation time,  $t_1$ , at which the parameter variance was less than  $6\sigma$  of the noise variance (additive). The upper plot shows the time in seconds and the bottom plot shows the time in terms of the number of periods. The maximum number of periods to sample was set to 100, and we see that some of the mid-frequencies, and at the higher frequencies, very little data was used in the estimate, thus accounting for the poor quality of the frequency response data.

With a one band fit, we were able to obtain a 85th order stable system with a normalized error of 2.167. We used a pre-warp term of 90 radians. This estimate was not improved with the multi-band method.

#### 4.4 Example 4: Fitting Real Frequency Response Data from a Stable System

Certainly, a speaker is not the highly resonant, flexible structure we have originally designed this method for. But can we expect this method to work well for all stable structures?

The experimental data is illustrated in Figure 15. Using the original SK algorithm, the highest stable fit was obtained with 76 states and an error norm of 2.1884. Recall that the algorithm uses the fraction  $\frac{1}{|d(s)^2|}$  rather than  $\frac{1}{|d(s)|}$  which is the default for our code (see Section 2.2.3).

The percentage error between the data and estimate is also shown in Figure 16. We obtained a 113th order system, with cutoff frequencies at 5.3 Hz and 87.5 Hz. The warp term for the first band was 3 Hz, and for the second band

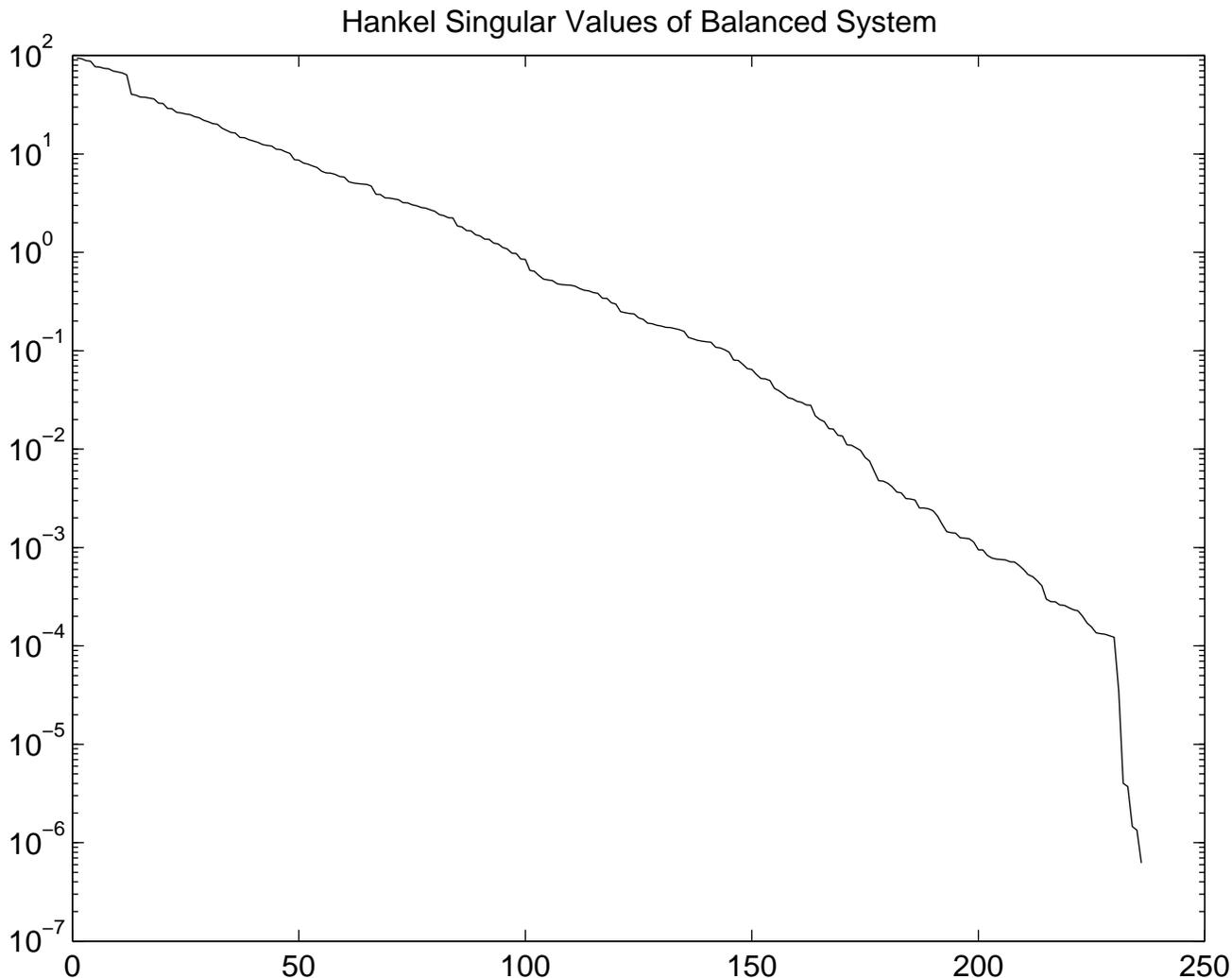


Figure 9: Hankel Singular Values.

it was 40 Hz. The normalized error after 10 iterations was 0.407.

Balancing the estimate did not yield any reduction in the number of states of the estimate. We were able to truncate the system to 104 states and maintain an error less than  $2e-4$ . However, since some of the data at the low frequencies was already on the order of  $1e-4$ , this error could be considered unacceptable.

Table 3 shows the relative performance on this data using a single-band fit. Note that even though the error norm is fairly small with a pre-warp term of 700 rad/second, the performance of the fit at low amplitudes (notably at the low frequencies is poor). This table illustrates the general benefit that can be gained from mapping the data onto a specific portion of the unit circle. Two of the implementation issues we've discussed are illustrated here: the use of  $d^i(s)$  as a weighting function, and the pre-warp term. Note that the use of both of these techniques provides a higher order fit *and* a smaller error (in the 2-norm sense). Of course, with such a high dynamic range, this one-band estimate achieves the lower error at the cost of ignoring the low magnitude data in the low frequencies.

## 5 Conclusions

Frequency domain system identification of high-order, lightly-damped, stable structures, suitable for control applications, is facilitated through the combination of the following key ideas:

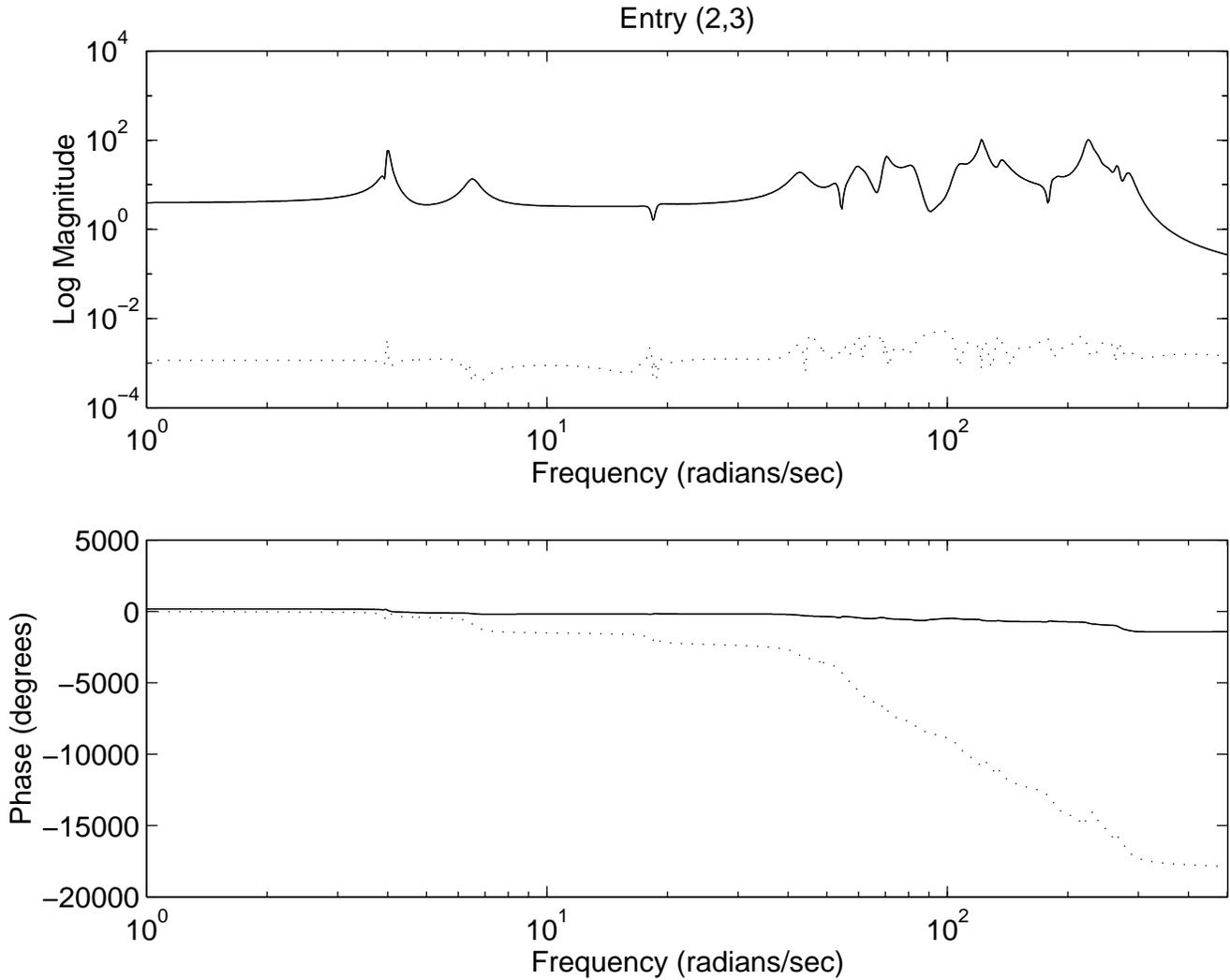


Figure 10: Comparison of Truncated (180 states) and Full Estimate (236 states), (2,3).

Weight	default pre-warp		user pre-warp	
	84.97 rad/sec	norm error	700 rad/sec	norm error
$\frac{1}{d(s)}$	76	2.18	127	0.19
$\frac{1}{\sqrt{d(s)}}$	100	2.09	149	0.17

Table 3: Highest Order Stable Estimates and Error, Single-Band Fit.

- Re-Shape the system (MIMO) as a one input system (SIMO),
- Bilinear Transform - map the continuous-time data to a favorable region in the discrete domain using the pre-warp term,
- Santhanum-Koerner procedure - use iterative least-squares method,
- Composite curve fitting - divide the data into sub-bands,
- Realize the original MIMO system from the estimate.

Further work is desired to determine analytic means or algorithms whereby this system estimation process can be automated. Areas where such study would be beneficial include:

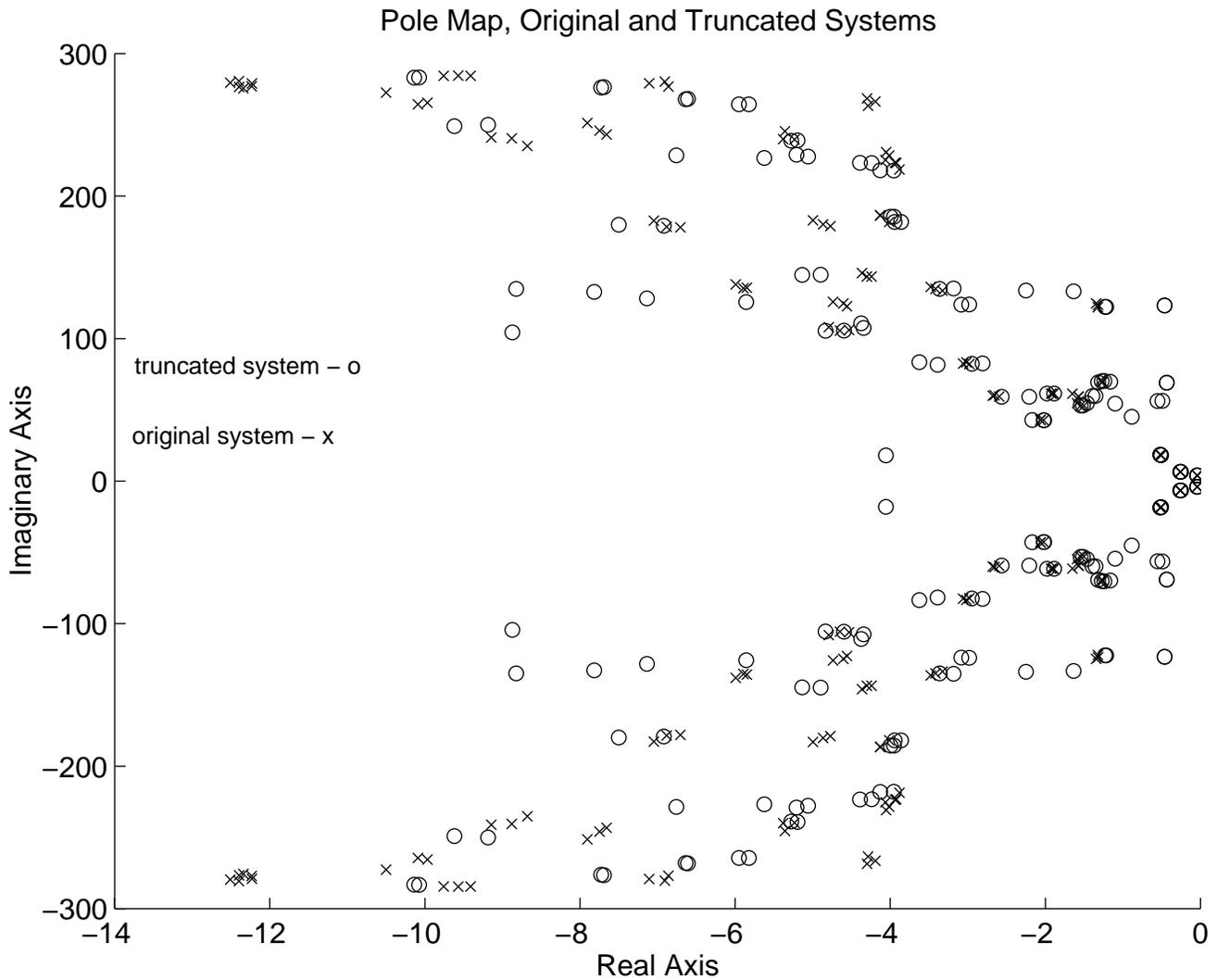


Figure 11: Comparison of Poles, Original and Truncated System.

- Optimal cut-off frequency selection,
- Determining lower and upper error bounds for a set of cut-off frequencies and model orders, and
- Automated optimization of the pre-warp term.

Methods for further refinement of the resulting estimate, including time-domain refinement and parametric refinement are available.

## References

- [1] David. S. Bayard, "High-Order Wide-Band Frequency Domain Identification using Composite Curve Fitting," *American Control Conference*, pp. 3181-5, 1992.
- [2] David. S. Bayard, "An Algorithm for State-Space Frequency Domain Identification Without Windowing Distortions," *Proceedings of the 31st Conference on Decision and Control*, pp. 1707-12, 1992.
- [3] D. S. Bayard, F. Y. Hadaegh, Y. Yam, R. E. Scheid, E. Mettler and M. H. Milman, "Automated On-orbit Frequency Domain Identification for Large Space Structures," *Automatica*, vol. 27, No. 6, pp. 931-46, 1991.

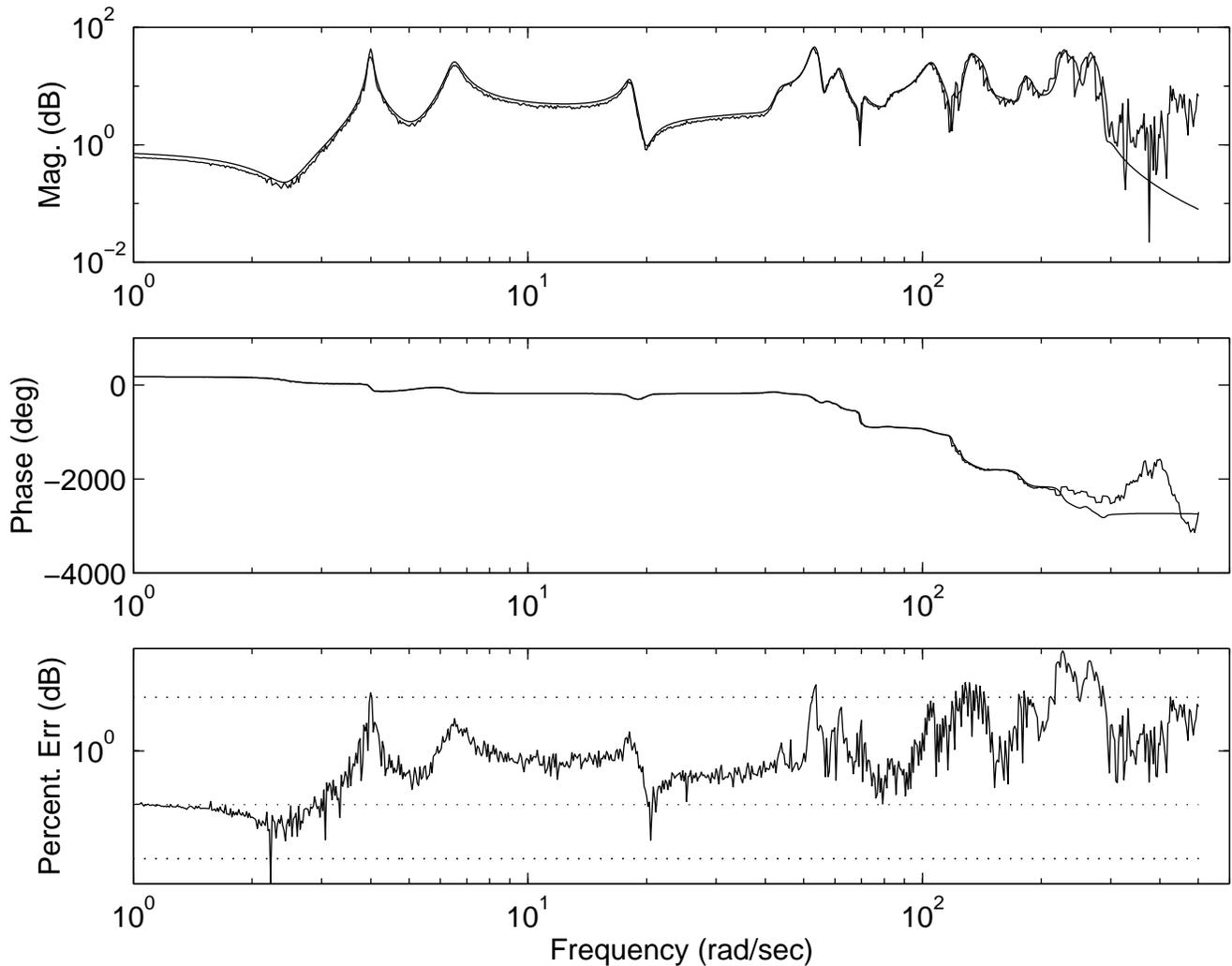


Figure 12: Actual and Simulated Frequency Response and Percentage Error.

- [4] David S. Bayard “Multivariable Frequency Domain Identification via 2-Norm Minimization,” *American Control Conference*, pp. 1253-1257, 1992.
- [5] D. S. Bayard, “Statistical Plant Set Estimation using Schroeder-Phased Multisinusoidal Input Design,” *American Control Conference*, pp. 2988-95, 1992.
- [6] Jean-Noel Aubrun, Alain C. Carrier, Kenneth R. Lorell, “Active Control for Segmented Mirror Optical Systems,” *Control and Dynamic Systems*, Vol. 57, 1993.
- [7] Alain Carrier, Rick Baldwin, Marshall Edwards, Richard Smith, “Principal Gain Tracking for Testing and System Identification with Application to the SPICE Testbed”, provided by the author.
- [8] Alain Carrier, Jean-Noel Aubrun, “Modal Characterization of the ASCIE Segmented Optics Test Bed: New Algorithms and Experimental Results,” provided by the author.
- [9] Young Man Cho, Guanghan Xu, T. Kailath, “Fast Identification of State-Space models via Exploitation of Displacement Structure,” *IEEE Transactions on Automatic Control*, vol. 39, No. 10, pp. 2004-2017, 1994.
- [10] M. Dahleh and J. B. Pearson, “ $\ell^1$ -Optimal controllers for MIMO discrete-time Systems,” *IEEE Trans. Automatic Control*, vol. AC-32, pp. 314-323, 1987.
- [11] J.E. Dennis, Jr. and R. B. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Prentice-Hall, 1983.
- [12] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, “State-space solutions to standard  $H_2$  and  $H_\infty$  control problems,” *IEEE Trans. Automat. Contr.*, vol. AC-34, no. 8, pp. 831-847, 1989.

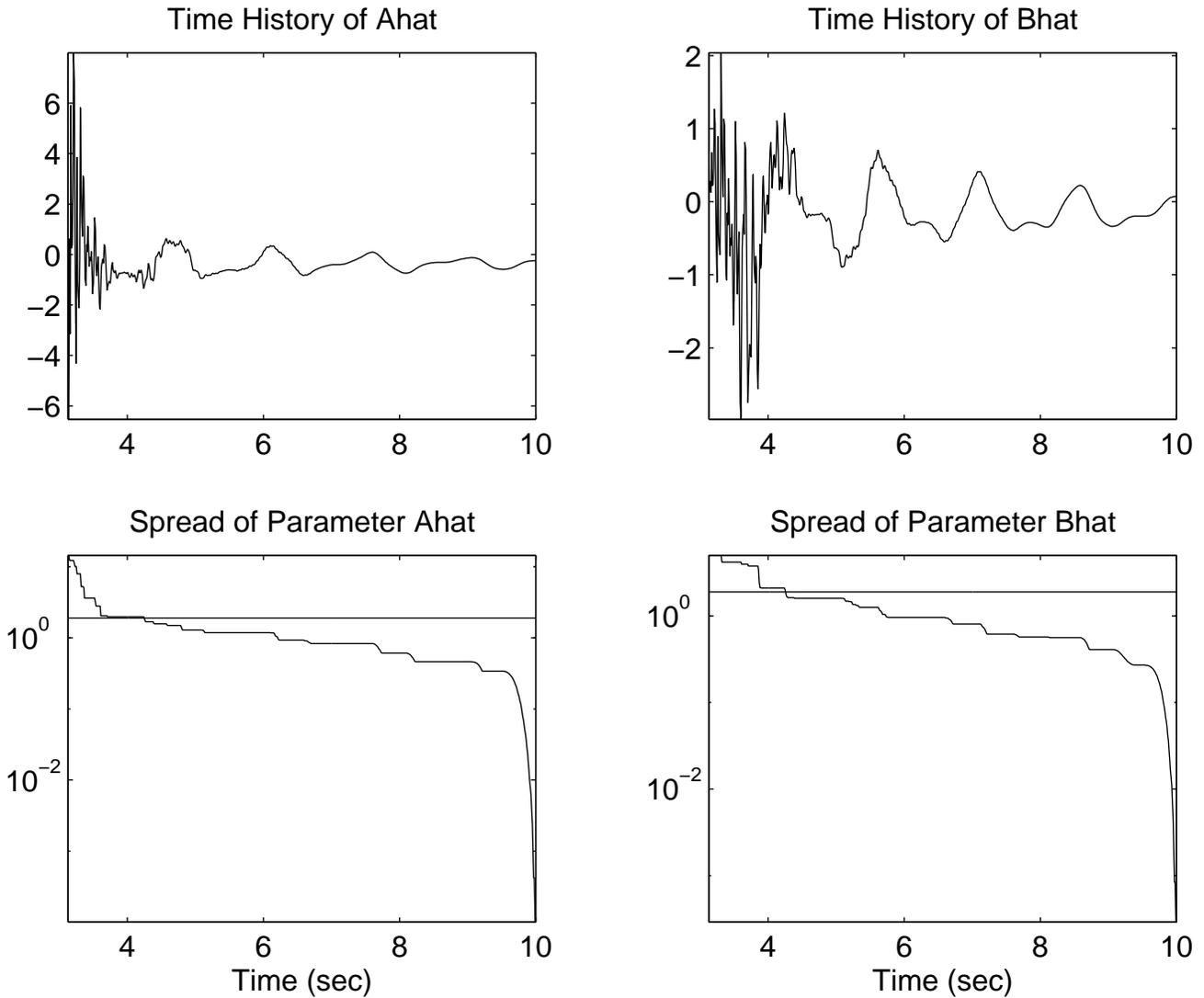


Figure 13: Time History of Parameter Estimates.

- [13] Darin Fisher, Deborah Jue, Andrew Packard, Kameshwar Poolla, "On the Identification of High-order Lightly-damped Multivariable Systems," submitted to ACC, 1998.
- [14] Jonathan H. Friedman, "Identification, Modeling, and Control of Flexible Structures," Ph.D. Dissertation, University of Michigan, 1996.
- [15] P. E. Gill, W. Murray, and M. H. Wright, *Practical optimization*, Academic Press, 1981.
- [16] L. Ljung, *System Identification, Theory for the User*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1987.
- [17] A. Packard and J.C. Doyle, "The complex structured singular value," *Automatica*, vol. 29, no. 9, pp:71-110, 1993.
- [18] Sundeep Rangan, "Stochastic  $H_\infty$  Identification: An Iteratively Weighted Least Squares Approach," M.S. Thesis, University of California, 1994.
- [19] C. Santhanum, J. Koerner, "Transfer function synthesis as a ratio of two complex polynomials," *IEEE Transactions on Automatic Control*, vol. 8, pp. 56-58, 1963.
- [20] M. Verhaegen, "Identification of the Deterministic Part of MIMO State Space Models given in Innovations Form from Input-Output Data", *Automatica*, vol 20, No. 1, pp. 61-74, 1994.
- [21] Bo Wahlberg, "System Identification using Laguerre Models," *IEEE Transactions on Automatic Control*, vol. 36, No. 5, pp. 551-561, 1994.

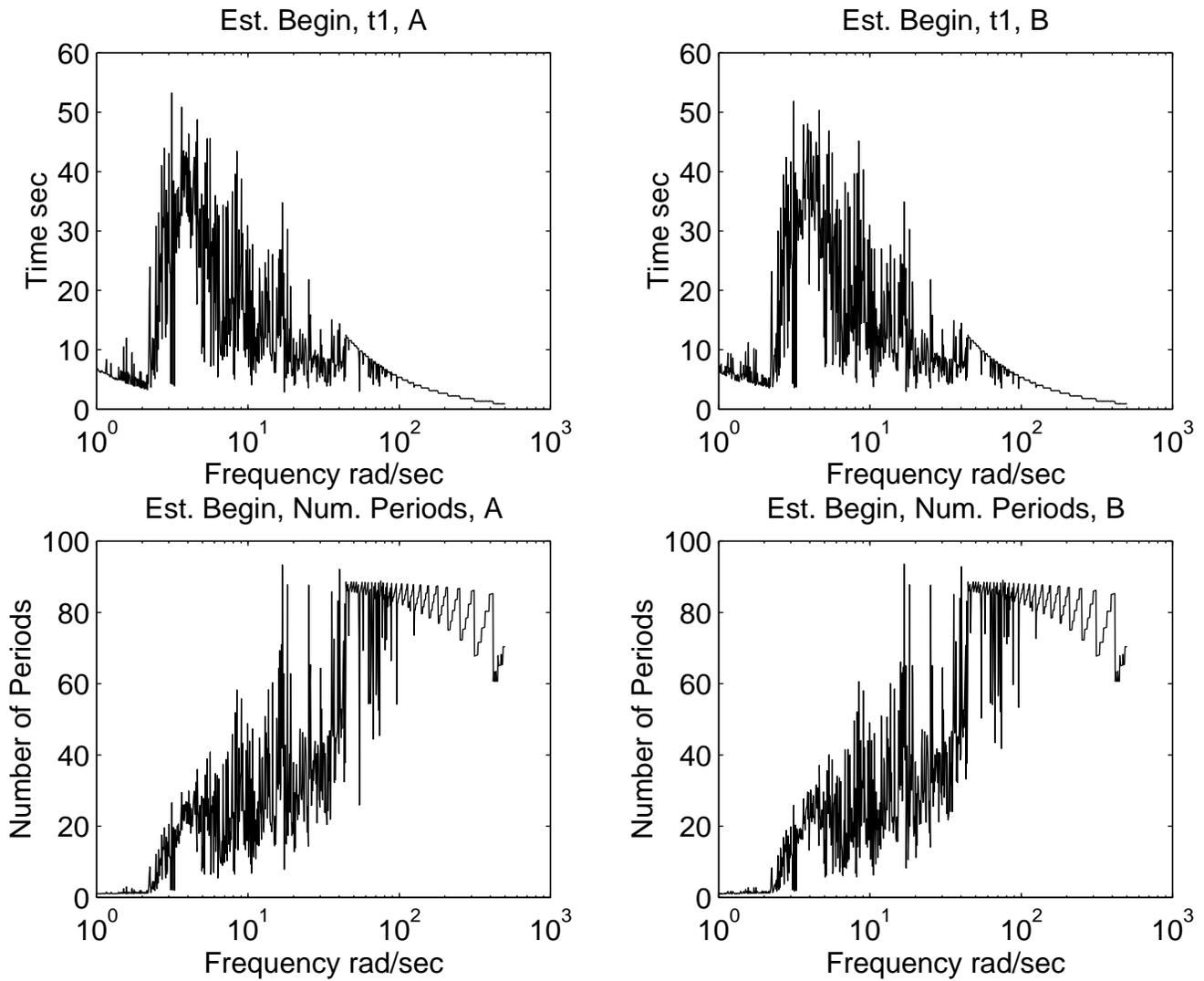


Figure 14: Time at Which Parameter Variance Dropped Below the Noise ( $6 \cdot \sigma$ ).

- [22] G. Wolodkin, S. Rangan, and K. Poolla, "An LFT approach to parameter estimation," *Proc. of the Automatic Control Conference*, San Francisco, pp. 2088-2092, June 1997.

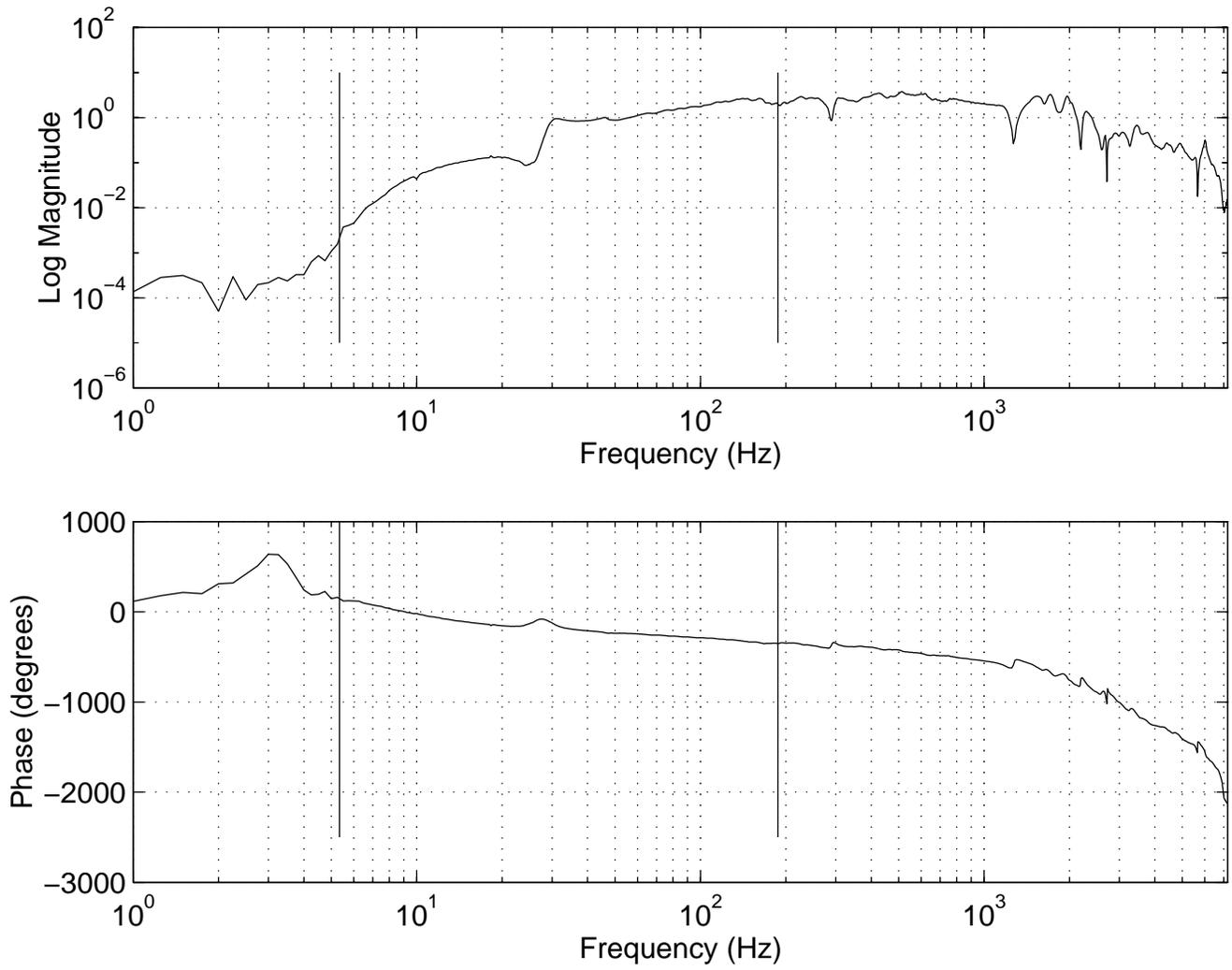


Figure 15: Measured Frequency Response Data from a Woofer.

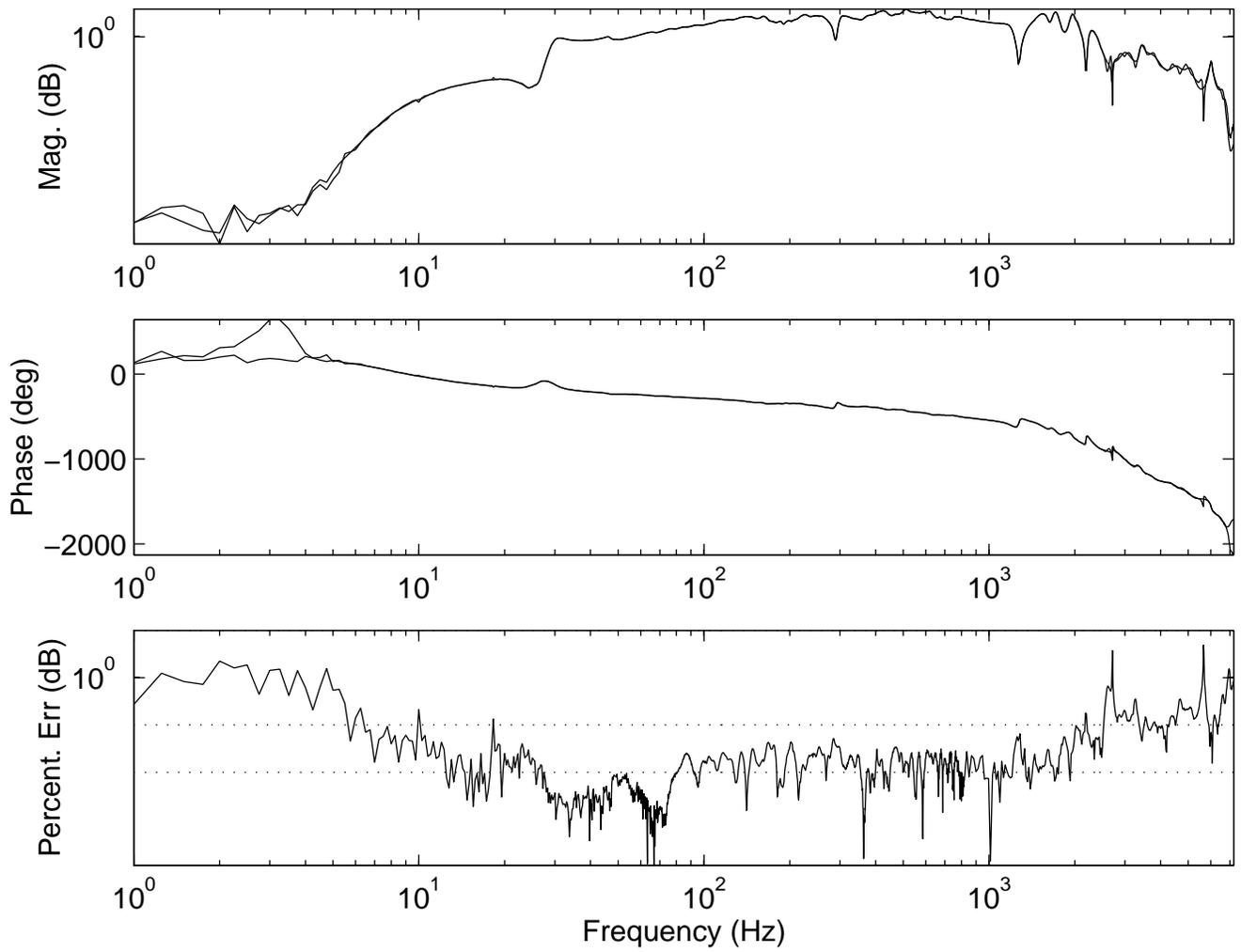


Figure 16: Comparison of Data and Estimate.